UNIVERSIDAD POLITÉCNICA
DE MADRID

FACULTAD DE INFORMATICA

MÁSTER UNIVERSITARIO EN INGENIERÍA DEL SOFTWARE –
EUROPEAN MASTER ON SOFTWARE ENGINEERING

Development of a Model for Security and Usability

Master Thesis

Laura Zapata Aspiazu

Madrid, July 2013

This thesis is submitted to the Facultad de Informática at Universidad Politécnica de Madrid in partial fulfillment of the requirements for the degree of Master of Science on Software Engineering.

Master Thesis
Master Universitario en Ingeniería del Software – European Master on Software Engineering
Thesis Title: Development of a Model for Security and Usability
Thesis no: EMSE-2013-04
July 2013

Author: Laura Zapata Aspiazu
Bachelor of Engineering in Computer Science
Escuela Superior Politécnica del Litoral, Ecuador

Supervisor:

Ana Moreno Sánchez-Capuchino
Full Professor

Languages, Systems and Software
Engineering Department
Facultad de Informática

Universidad Politécnica de Madrid

Co-supervisor:

Eduardo Fernández-Medina
Full Professor

Department of Information
Technologies and Systems
Escuela Superior de Informática

Universidad de Castilla-La Mancha

Facultad de Informática
Universidad Politécnica de Madrid
Campus de Montegancedo, s/n
28660 Boadilla del Monte (Madrid)
Spain

Contents

List of Figures

List of Tables

Abstract

This research addressed the development of a consolidated model designed especially to cover the security and usability attributes of a software product.

As a starting point, we built a new usability model on the basis of well-known quality standards and models. We then used an existing security model to analyse the relationship between these two approaches.

This analysis consisted of a systematic mapping study of the relationship between security and usability as global quality factors. We identified five relationship types: inverse, direct, relative, one-way inverse, and no relationship. Most authors agree that there is an inverse relationship between security and usability. However, this is not a unanimous finding, and this study unveils a number of open questions, like application domain dependency and the need to explore lower-level relationships between attribute subcharacteristics.

In order to clarify the questions raised during the research, we conducted a second systematic mapping to further analyse the finer-grained structure of these factors, such as authentication as a subset of security and user efficiency as a subset of usability. The most relevant finding is that efficiency does not depend on the security level during the authentication process.

There are other subfactors that require analysis. Accordingly, this research is the first part of a larger project to develop a full-blown consolidated model for security and usability.

1 Introduction

1.1 Introduction

Since many businesses are critically reliant on their information systems for key business processes, security has become a very important area for protecting data and information at risk from human and technical errors, accidents and disasters, fraud, commercial espionage, malicious damage and other threats, as reported by Information Communication Technology (ICT, undated).

On the other hand, according to (Braz et al., 2007), secure systems also need to be usable. If a highly secure system is unusable, users are likely to move their data to less secure but more usable systems, as reported by the Workshop on Usable Security (WEIS, 2012). Additionally, the same institution (WEIS, 2012) stated that "Problems with usability are a major contributor to many high-profile security failures today".

To many engineers, usability is synonymous with user interface design. However, usability is more than just about designing interfaces, it is a quality attribute concerning the people interacting with these interfaces and how they use them to perform tasks (Faily & Flechais, 2010). Another claim, stated by (Payne & Edwards, 2008), is that "the difference between a poor interface and a good interface can influence users' ability to perform tasks securely". In this context, there is increasing agreement that we need to design secure systems that people can actually use, but less agreement about how to do this (Cranor & Garfinkel 2005). A field called human-computer interaction and security (HCISec) has come into being in order to try to solve this conflict and to make a trade-off between security and usability, (Malenkovich 2012). Consequently, current literature is replete with major debates concerning this balance. In this context, Tobias Hahn et al. (2012) claim that "Here, user-friendliness goes too far at the cost of security". In contrast, (Prakash, 2007) claims that "… studies also identified gaps in security in real-world systems due to inadequate interface design". These examples represent contrary points of view. On this ground, this research aims to look into the exact link between security and usability.

As with other products, software system quality can be quantified, and quality standards have been created to do this. There are currently a great many quality standards, and older versions have even been improved leading to the development of new models.

The latest quality standard is ISO/IEC 25010, also called SQuaRE, which defines the security attribute in terms of its subattributes: confidentiality, integrity, non-repudiation, accountability, security compliance and authenticity. This standard also defines the usability attribute in terms of its subattributes: effectiveness, efficiency, satisfactions and usability compliance.

Based on these definitions, the purpose of this research is to study the detailed link between security and usability as quality attributes, and explore the link between their subattributes.

We set out to answer the following question: Does security increase or decrease with usability? If we identify this relationship, we should be able to predict usability measures for a given security measure.

Nevertheless, the number of possible combinations of security and usability subattributes is huge. Consequently, we have not yet fully developed the consolidated model for security and usability, because there are other subfactors that remain to be analysed. Indeed, the reported research represents the first part of a larger project.

1.2 Objectives

The objectives of this research follow:

- ✓ Define a consolidated security model based on the different security classifications and definitions.
- ✓ Define a consolidated usability model based on the different usability classifications and definitions.
- ✓ Explore the relationship between security and usability at attribute level.
- ✓ Explore the relationship between representative security and usability subattributes.

1.3 Document Structure

This document has the format of a multiple-paper thesis. Chapters 2 to 5 are each self-contained (in the sense that they can each be read and understood separately). However, we present general conclusions related to the research topic as a whole.

The document is structured as follows. Chapter 2 outlines the definitions of a consolidated security model based on the different security classifications and definitions. Chapter 3 outlines the definitions of a consolidated usability model based on the different usability classifications and definitions. Chapter 4 explores the relationship between security and usability at attribute level. Chapter 5 explores the relationship between representative security and usability subattributes. Chapter 6 presents general conclusions regarding the research topic.

References

- Braz, C., Seffah, A., M'Raihi, D. (2007). Designing a Trade-off between Usability and Security: A Metrics Based-Model. Springer volume 4663, 114-126.
- Cranor Lorrie Faith, Garfinkel Simson. (2005). Security & Usability, Designing Secure Systems that People Can Use, O'Reilly Media.
- Faily Shamal, Flechais Ivan. (2010). *Analysing and Visualising Security and Usability in IRIS.* 2010 International Conference on Availability, Reliability and Security.
- Kaspersky Lab. Available from: <http://blog.kaspersky.com/usability-and-security-the-endless-pursuit-of-perfection/>. [09 June 2013].
- Malenkovich Serge (2012). Usability and Security: *The endless pursuit of perfection.*
- WEIS - Workshop on Usable Security 2012. ISE Information Security Economics. Available from: <http://infosecon.net/usec12/index.php>. [14 June 201].
- Information Communication Technology (ICT). Available from: <http://www.tutor2u.net/business/ict/intro_security_introduction.htm>. [14 June 2013].
- Payne Bryan D., Edwards W. Keith. (2008) Georgia Institute of Technology. IEEE Computer Society.

2 Security Model

We are working in partnership with Castilla-La Mancha University, which put their security quality model at our disposal. Its construction is detailed in the document titled *Modelo de Calidad para la Seguridad*, produced as part of the Medusas project (Sicaman Nuevas Tecnologías 2012).

2.1 Description

This model specialised in security was tailored using well-known quality standards and models, such as SQuaRE, the Firesmith model, COBIT, MAGERIT and others.

Briefly, this model can be said to focus on two major issues. The first is protection against information and data security threats, and the second is related to safety issues. Each part is composed of a set of characteristics that are further subdivided into subcharacteristics. Note also that this model is enriched with more security-specific aspects than existing quality standards, which measure the overall quality.

As we did not author this model, we cannot unveil its structure, but we do have permission to describe some of the parts that we will use during this research, like, for instance, the *authenticity* characteristic, which is defined as the degree to which a subject or resource can prove that they are who or what they claim to be. This characteristic is composed of the *authentication* and *identification* subcharacteristics.

Authentication is defined as the degree to which the system verifies the identities of its externals before interacting with them. On the other hand, *identification* is defined as the degree to which the system identifies (i.e., recognizes) its externals before interacting with them.

References

- Sicaman Nuevas Tecnologías. (2012). Modelo de Calidad para la Seguridad. Proyecto Medusas (Mejora y Evaluación del Diseño, Usabilidad, Seguridad y Mantenibilidad del Software).

3 Usability Model

3.1 Introduction

It is no longer sufficient to just deliver technically excellent systems. There is a growing demand for computer systems that are widely accessible, easy to learn and use, easy to integrate into work or leisure activities and, at the same time, perform efficiently. Despite the rapid increase in computer power and more sophisticated systems development, these objectives are not being achieved: it is widely believed that most computer users still cannot get their systems to do exactly what they want, as stated by Bevan (1999).

In order to produce systems that are better matched to user needs, it is essential to enhance current development processes to obtain high quality software products. The quality of a system product can be defined as the degree to which the system satisfies the stated and implied needs of all of its stakeholders, as defined by the ISO/IEC 25010 (ISO/IEC, 2009). These needs are specified by a wide variety of models, standards, guidelines or characteristics that can be used to measure and evaluate the quality of a system.

Whereas some models have a similar structure, others establish their own parameters that sometimes use different terminology to mean the same thing, others again use parameters that are not considered by the rest, and the worst case is when other standards use terminology that change the meaning of the attribute. This is what has led us to study the best-known quality standards and models in order to build a new, understandable model using suitable terminology in order to consolidate a new model from a combination of existing approaches.

3.2 Usability In Quality Standards

There are many models and frameworks that evaluate the quality of a software product, the quality in use of a system and the quality of the data. Almost all of these models include usability. This section introduces some of the best-known standards focusing on the part of the model that describes system usability.

3.2.1 ISO 9241-11

The ISO 9241-11(1998) standard, titled *Guidance on usability,* is Part 11 of the 1998 ISO 9241 standard that was originally titled *Ergonomic requirements for office work with visual display terminals (VDTs)*. ISO 9241 was renamed *Ergonomics of human system interaction* in 2006, and since then ISO has been

working on renaming some its parts in order to cover more human-computer interaction topics. As Part 11 has not yet changed (as emphasized in IEEE Standards Status Report & ISO Catalogue) we will use the 1998 version in this research.

The standard addresses *usability* as a factor which is further subdivided into three subfactors: *effectiveness*, *efficiency* *and* *satisfaction*. This factor and its respective subfactors are defined as follows.

- **Usability** is the extent to which a product can be used by specified users to achieve specific goals with effectiveness, efficiency and satisfaction in a specified context of use[1].
  - **Effectiveness** is the accuracy and completeness with which users achieve specified goals.
  - **Efficiency** is the resources expended in relation to the accuracy and completeness with which users achieve goals.
  - **Satisfaction** is the freedom from discomfort, and positive attitudes towards product use.

The hierarchy of factor and subfactors extracted from this standard is shown in Table 3.1:

Table 3.1 Usability in ISO standard 9241-11

Characteristic	Subcharacteristic
Usability	Effectiveness
	Efficiency
	Satisfaction

The approach of this model can be briefly said to focus on two general-purpose categories: *performance* and *satisfaction. Performance* (effectiveness and efficiency) in product usage is an objective attribute, whereas *satisfaction* is subjective because it is peculiar to each user.

This standard explains a framework for specifying usability in terms of *usability measures* (effectiveness, efficiency and satisfaction) with respect to *goals* established over the product *context of use*. In order to measure usability it is necessary to identify the goals and to decompose effectiveness, efficiency and satisfaction and the components of the context of use into subcomponents with measurable and verifiable attributes.

[1] **Context of use:** Users, tasks, equipment (hardware, software and materials), and the physical and social environments in which a product is used.

Usability measures of effectiveness, efficiency and satisfaction can be specified for overall goals or for narrower goals. Examples of appropriate measures are given in Table 3.2, (ISO 9241-11, 1998).

Table 3.2 ISO 9241-11 examples of overall usability measures

Usability objective	Effectiveness measures	Efficiency measures	Satisfaction measures
Overall usability	Percentage of goals achieved Percentage of users successfully completing task Average accuracy of completed tasks	Time to complete a task Tasks completed per unit time Monetary costs of performing the task	Rating scale for satisfaction Frequency of discretionary use Frequency of complaints

Additional measures may be required for particular target product properties that improve usability. Examples of some of these properties and additional specialised measures are given in Table 3.3, (ISO 9241-11, 1998).

Table 3.3 ISO 9241-11 usability for a particular property

Usability objective	Effectiveness measures	Efficiency measures	Satisfaction measures
Satisfaction of needs of trained users	Number of power tasks performed; Percentage of relevant functions used	Relative efficiency compared with an expert user	Rating scale for satisfaction with power features
Satisfaction of needs to walk up and use	Percentage of tasks completed successfully on first attempt	Time taken on first attempt; Relative efficiency on first attempt	Rate of voluntary use
Satisfaction of needs for infrequent or intermittent use	Percentage of tasks completed successfully after a specified period of non-use	Time spent re-learning functions; Number of persistent errors	Frequency of use
Minimization of support requirements	Number of references to documentation; Number of calls to support; Number of accesses to help	Productive time; Time to learn to criterion	Rating scale for satisfaction with support facilities
Learnability	Number of functions learned; Percentage of users who manage to learn to criterion	Time to learn to criterion; Time to re-learn to criterion; Relative efficiency while learning	Rating scale for ease of learning
Error tolerance	Percentage of errors corrected or reported by the system; Numbers of user errors tolerated	Time spent on correcting errors	Rating scale for error handling
Legibility	Percentage of words read correctly at normal viewing distance	Time to correctly read a specified number of characters	Rating scale for effort

3.2.2 ISO/IEC 9126

ISO/IEC 9126 (2001), titled *Software engineering – Product quality,* is a multi-part software quality product standard. It consists of the following parts:

- Part 1: Quality model, ISO/IEC 9126-1
- Part 2: External metrics, ISO/IEC 9126 2
- Part 3: Internal metrics, ISO/IEC 9126 -3
- Part 4: Quality-in-use metrics, ISO/IEC 9126 -4

This standard replaced the original ISO/IEC 9126:1991, but it was itself replaced by SQuaRE in September 2012. Although it was withdrawn in 2012, it is still one of the most widespread software product quality standards, on which ground it is included in this research.

3.2.2.1 ISO/IEC 9126-1

The ISO/IEC 9126-1 (2001) standard, titled *Quality model,* describes a two-part model for software product quality:

- Internal quality and external quality, and
- Quality in use.

The first part of the model specifies six characteristics for *internal and external quality*: functionality, reliability, efficiency, maintainability, portability and usability (which are further subdivided into subcharacteristics). As we can see from this model, *usability* is defined as a characteristic, which is subdivided into *understandability*, *learnability*, *operability*, *attractiveness* and *usability compliance.*

The internal and external quality model defines the usability quality factor and its respective subfactors as follows.

- **Usability** is a set of attributes that bear on the effort needed for use and on the individual assessment of such use by a stated or implied set of users:
  - **Understandability** is the capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use.
  - **Learnability** is the capability of the software product to enable the user to learn its application.
  - **Operability** is the capability of the software product to enable the user to operate and control it.
  - **Attractiveness** is the capability of the software product to be attractive to the user.
  - **Usability Compliance** is the capability of the software product to adhere to standards, conventions, style guides or regulations relating to usability.

The second part of the model specifies four *quality-in-use* characteristics: *effectiveness, productivity, safety,* and *satisfaction*, but does not further specify the quality-in-use model beyond the level of characteristics.

The definitions that the standard gives for the quality-in-use model are as follows.

- **Effectiveness** is the capability of the software product to enable users to achieve specified goals with accuracy and completeness in a specified context of use.
- **Productivity** is the capability of the software product to enable users to expend appropriate amounts of resources in relation to the effectiveness achieved in a specified context of use.
- **Safety** is the capability of the software product to achieve acceptable levels of risk of harm to people, business, software, property or the environment in a specified context of use.
- **Satisfaction** is the capability of the software product to satisfy users in a specified context of use.

A summary of this model is shown in Table 3.4. Note that the internal and external quality model shown in Table 3.4 is confined to the usability characteristic and subcharacteristics only, whereas the quality-in-use model covers all characteristics, because, as mentioned above, this standard does not further specify the model for quality in use beyond the level of characteristics.

Table 3.4 ISO/IEC 9126 usability-related internal

Characteristic	Subcharacteristic
Internal and External Quality Model	
Usability	Understandability
	Learnability
	Operability
	Attractiveness
	Usability Compliance
Quality-in-use Model	
Effectiveness	
Productivity	
Safety	
Satisfaction	

3.2.2.2 ISO/IEC 9126-2, ISO/IEC 9126-3, ISO/IEC 9126-4

ISO/IEC 9126-1 (ISO/IEC, 2001) defines terms for the software quality characteristics and how these characteristics are decomposed into

subcharacteristics. ISO/IEC 9126-1 does not, however, describe how any of these subcharacteristics could be measured. ISO/IEC 9126-2 (ISO/IEC, 2003) defines external metrics, ISO/IEC 9126-3 (ISO/IEC, 2003) defines internal metrics and ISO/IEC 9126-4 (ISO/IEC, 2004) defines quality-in-use metrics for measuring the characteristics or the subcharacteristics.

Internal metrics measure the actual software, external metrics measure the behaviour of the computer-based system that includes the software, and quality-in-use metrics measure the effects of using the software in a specific context of use. Internal metrics are static measures that do not rely on software execution, whereas external metrics are applicable to running software (dynamic measures), as pointed out by (Bucur, 2006). Quality-in-use metrics are only applicable when the final product is used in real conditions (Colin et al., 2008).

Internal metrics may be applied to a non-executable software product (such as request for proposal, requirements definition, design specification or source code) during its development. Internal metrics provide users with the opportunity to measure the quality of the intermediate deliverables and thereby predict the quality of the final product. This allows users to identify quality issues and take corrective action as early as possible in the development life cycle.

External metrics may be used to measure the quality of the software product by measuring the behaviour of the system of which it is a part. External metrics can only be used during the testing stages of the life cycle process and during any operational stages. The measurement is performed when executing the software product in the system environment in which it is intended to operate.

Quality-in-use metrics measure whether a product meets the needs of specified users to achieve specified goals with effectiveness, productivity, safety and satisfaction in a specified context of use. This can be only achieved in a realistic system environment.

Table 3.5.a and Table 3.5.b show some examples of metrics that are applicable to a usability subcharacteristic of the internal and external quality model extracted from parts 2 and 3 of the model, (ISO/IEC 9126-2 & ISO/IEC 9126-3 , 2003).

Table 3.5.a ISO/IEC 9126 examples of internal and external metrics associated with usability.

	Internal metric		External metric	
	Name (purpose)	**Measurement**	**Name** (purpose)	**Measurement**
Understandability	**Completeness of description** (What proportion of functions is described in the product description?)	Number of functions described in the product description divided by total number of functions	**Completeness of description** (What proportion of functions is understood after reading the product description?)	Number of functions understood divided by total number of functions
Learnability	**Completeness of user documentation and/or help facility** (What proportion of functions is described in the user documentation and/or help facility?)	Number of functions described divided by total of number of functions provided	**Help frequency** (How frequently does a user have to access help to learn operation to complete his/her work task?)	Number of accesses to help until a user completes his/her task
Operability	**User operation undoability** (What proportion of functions can be undone?)	Number of implemented functions which can be undone by the user divided by number of functions	**Default value availability in use** (Can users easily select parameter values for convenient operation?)	Number of times users fail to establish or to select parameter values divided by total number of times that users attempt to establish or to select parameter values

Table3.5.b ISO/IEC 9126 examples of internal and external metrics associated with usability, (continuation Table 3.5.a)

	Internal metric		External metric	
	Name (purpose)	**Measurement**	**Name** (purpose)	**Measurement**
Attractiveness	**User interface appearance customizability** (What proportion of user interface elements can be customized in appearance?)	Number of types of interface elements that can be customized divided by total number of types of interface elements	**Interface appearance customizability** (What proportion of the appearance of interface elements can be customized to the user's satisfaction?)	Number of interface elements whose appearance is customized to user's satisfaction divided by number of interface elements that the user wished to customize
Usability compliance	**Usability compliance** (How compliant is the product to applicable regulations, standards and conventions for usability?)	Number of correctly implemented items related to usability compliance confirmed in evaluation divided by total number of compliance items	**Usability compliance** (How completely does the software adhere to the standards, conventions, style guides or regulations relating to usability?)	Number of specified usability compliance items that have not been implemented better during testing divided by total number of specified usability compliance items

Table 3.6 shows some examples of metrics that are applicable to characteristics of the quality-in-use model extracted from part 4 of the model, (ISO/IEC 9126-4, 2004).

Table 3.6 ISO/IEC 9126 examples of quality-in-use metrics

Quality-in-use metric		
Characteristic	**Name** (purpose)	**Measurement**
Effectiveness	**Task completion** (What proportion of the tasks are completed?)	Number of tasks completed divided by total number of tasks attempted
Productivity	**Productive proportion** (What proportion of the time is the user performing productive actions?)	Productive time divided by task time, where productive time = task time – help time – error time – search time
Safety	**Safety of people affected by system use** (What is the ratio of risk to people affected by system use?)	Number of people put at risk divided by total number of people potentially affected by the system
Satisfaction	**Satisfaction scale** (How satisfied is the user?)	Questionnaire producing psychometric scales divided by population average

The internal and external quality model is clear with respect to usability, whereas the quality-in-use model does not specify which of these characteristics are applicable to usability and is more vague.

3.2.3 SQuaRE

SQuaRE is a series of standards, titled *Software product Quality Requirements and Evaluation,* consisting of the following divisions:

- ISO/IEC 2500n - Quality Management Division,
- ISO/IEC 2501n - Quality Model Division,
- ISO/IEC 2502n - Quality Measurement Division,
- ISO/IEC 2503n - Quality Requirements Division,
- ISO/IEC 2504n - Quality Evaluation Division,
- ISO/IEC 25050 – 25099 SQuaRE extension standards.

The focus of this research is on the quality model division (ISO/IEC 25010).

3.2.3.1 ISO/IEC 25010

The ISO/IEC 25010 (ISO/IEC, 2011) standard, titled *Quality model division,* is derived from ISO/IEC 9126-1 (ISO/IEC, 2001), which it amends by using better names, raising subcharacteristics to the level of characteristics, and so on. Basically, however, it retains the same structure.
The standard describes a two-part model for software product quality:

- A *software product quality model* composed of eight characteristics that relate to static properties of the software and dynamic properties of the computer system.
- A *system quality-in-use model* composed of three characteristics that relate to the outcome of interaction when a product is used in a particular context of use.

ISO/IEC 25012 also contains a model for data quality that is complementary to this model. The data quality model is not concerned with usability, for which reason it is not described here.

The *software product quality model* categorizes software quality attributes into eight characteristics (functional suitability, reliability, performance efficiency, operability, security, compatibility, maintainability and portability). Each characteristic is composed of a set of related subcharacteristics. The subcharacteristics can be measured by internal or external measures.

The usability amendments (carried out to specify ISO/IEC 25010 which is derived from ISO/IEC 9126-1) are shown in Table 3.7.

Table 3.7 Amendments to ISO/IEC 9226-1 to create SQuaRE

ISO/IEC 25010 (SQuaRE)	ISO/IEC 9126-1	Notes
Operability	Usability	Renamed to avoid conflict with the definition of usability[2] in 25062
Appropriateness recognisability	Understandability	New name is more accurate
Learnability	Learnability	
Ease of use	Operability	Renamed
Attractiveness	Attractiveness	
Technical accessibility		New subcharacteristic

[2] Author (Bevan, 2009) explain the reason of this change, he stated that: In 2006 the US standard for a Common Industry Format for Usability Test Reports (CIF) was adopted by ISO as part of the revised Software product Quality Requirements and Evaluation (SQuaRE) set of standards. As potential users of the CIF had originally expressed a preference for the term *usability* rather than *quality in use*, the ISO 9241-11 definition of usability was retained when the CIF became part of this series. When the ISO/IEC 9126-1 quality model came to be incorporated into the SQuaRE series (as ISO/IEC 25010), some ISO/IEC national bodies commented on the discrepancy between the narrow definition of usability inherited from ISO/IEC 9126 and the broader definition in the CIF. But with the higher profile of usability in industry, there was now pressure to align the SQuaRE definition with the CIF, rather than vice versa. This was achieved by renaming the narrower ISO/IEC 9126 concept of usability as operability. This made it possible to define usability as a characteristic of quality in use, with subcharacteristics of effectiveness, efficiency and satisfaction.

This model establishes *operability* as a characteristic, which is divided into *appropriateness recognisability, learnability, ease of use, attractiveness, technical accessibility and operability compliance.*

The standard defines operability and its subcharacteristics as follows.

- **Operability** is the degree to which the product has attributes that enable it to be understood, learned, used and attractive to the user[3], when used under specified conditions. Users may include operators, end users and indirect users who are under the influence of or dependent on the use of the software.
  - **Appropriateness recognisability** is the degree to which the product provides information that enables users to recognize whether the software is appropriate for their needs.
  - **Learnability** is the degree to which the product enables users to learn its application.
  - **Ease of use** is the degree to which users find the product easy to operate and control.
  - **Attractiveness** is the degree to which the product is attractive to the user.
  - **Technical accessibility** is the degree to which users with specified disabilities can operate the product.
  - **Operability compliance** is the degree to which the product adheres to standards, conventions, style guides or regulations in laws and similar prescriptions relating to operability.

[3] **Users** may include operators, end users and indirect users who are under de influence of or dependent on the use of the software.

The second part of the model *system quality-in-use model* defines quality in use as the degree to which a product used by specific users meets their needs to achieve specific goals with effectiveness, efficiency, flexibility, safety and satisfaction in specific context of use.

The standard defines three characteristics: usability, flexibility and safety. The subcharacteristics of *usability* are *effectiveness*, *efficiency*, *satisfaction* and *usability compliance*.

The standard defines the usability characteristic and its subcharacteristics as follows.

- **Usability** is the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use:
- **Effectiveness** is the accuracy and completeness with which users achieve specified goals.
- **Efficiency** is the resources expended in relation to the accuracy and completeness with which users achieve goals.
- **Satisfaction** is the degree to which users are satisfied in a specific context of use. It can be further subdivided into the following subcharacteristics:
  - **Likability** is cognitive satisfaction.
  - **Pleasure** is emotional satisfaction.
  - **Comfort** is physical satisfaction.
  - **Trust** is satisfaction with security.
- **Usability compliance** is the degree to which the product adheres to standards or conventions relating to usability.

Table 3.8 summarizes this model related to *operability* in the *software product quality model and usability* in the *system quality-in-use model*.

This model improved ISO/IEC 9126-1 especially with respect to the structure of the quality-in-use model, because it incorporates a hierarchy of characteristics and subcharacteristics that the ISO/IEC 9126-1 does not have.

Table 3.8 SQuaRE, 25010 part: quality model division

Characteristic	Subcharacteristic	Sub-subcharacteristics
Software Product Quality Model		
Operability	Appropriateness recognisability	
	Learnability	
	Ease of use	
	Attractiveness	
	Technical accessibility	
	Operability compliance	
System Quality-in-Use Model		
Usability	Effectiveness	
	Efficiency	
	Satisfaction	Likability
		Pleasure
		Comfort
		Trust
	Usability compliance	

3.2.3.2 ISO/IEC 25022, ISO/IEC 25023 and ISO/IEC 25024

As in Section 3.2.2.2, the aim of this section was to build a table with some examples of metrics for a better understanding of the model. SQuaRE standard has the 2502n division for quality measures, which is composed of:

- 25022: Internal metrics replaces ISO/IEC 9126-3
- 25023: External metrics replaces ISO/IEC 9126-2
- 25024: Quality-in-use measures replace ISO/IEC 9126-4.

But at the time of writing, these documents had not been released.

3.2.4 Quality-In-Use Integrated Measurement (QUIM)

Besides the standards and models described in the previous sections, some models have been proposed by other authors. This section describes QUIM, since this is a model is widely accepted in the literature.

QUIM (Quality-in-Use Integrated Measurement) is a consolidated model for usability measurement. QUIM is hierarchical and decomposes usability into factors, then into criteria, and finally into specific metrics, (Seffah et al., 2006).

QUIM is a consolidated model that takes the ISO 9241-11 standard as a baseline and adds some other characteristics from ISO 9126. Motivated by special kinds of users, such as disabled persons, QUIM also includes the universality and accessibility factors, and it also includes safety (human security) as a usability characteristic applicable for some sorts of critical system.

Table 3.9 summarizes QUIM's ten usability factors.

Table 3.9 QUIM Factors

	Factors
Usability	Efficiency
	Effectiveness
	Productivity
	Satisfaction
	Learnability
	Safety
	Trustfulness
	Accessibility
	Universality
	Usefulness

A description of QUIM's factors follows (Seffah et al., 2006):

1. **Efficiency** is the capability of the software product to enable users to expend appropriate amounts of resources in relation to the effectiveness achieved in a specified context of use.
2. **Effectiveness** is the capability of the software product to enable users to achieve specified tasks with accuracy and completeness.
3. **Productivity** is the level of effectiveness achieved in relation to the resources (i.e., time to complete tasks, user efforts, materials or financial cost of usage) consumed by the users and the system. Unlike efficiency, productivity is concerned with the amount of useful output that is obtained from user interaction with the software product. Therefore, the definition of productivity

considers the productive resources that are expended in order to achieve tasks.

4. **Satisfaction** refers to the subjective responses from users about their feelings when using the software (i.e., is the user satisfied or happy with the system?).
5. **Learnability** is the ease with which the features required for achieving particular goals can be mastered. It is the capability of the software product to enable users to feel that they can productively use the software product right away and then quickly learn other new (for them) functionalities.
6. **Safety** is concerned with whether a software product limits the risk of harm to people or other resources, such as hardware or stored information.
7. **Trustfulness** is the reliability that a software product offers its users. This concept is perhaps most applicable to e-commerce websites, but it could potentially apply to many different kinds of software products.
8. **Accessibility** is the capability of a software product to be used by persons with some sort of disability (e.g., visual, hearing, psychomotor impairment).
9. **Universality** is concerned with whether a software product accommodates a diversity of users with different cultural backgrounds (e.g., local culture is considered).
10. **Usefulness** is whether a software product enables users to acceptably solve real problems. Usefulness implies that a software product has practical utility, which partly reflects how closely the product supports the user's own task model. Usefulness obviously depends on the features and functionality offered by the software product. It also reflects the knowledge and skill level of the users while performing some task (i.e., it does not consider just the software product).

3.3 Study of Characteristics

This section describes the steps followed to build the usability model which is the aim of this research. We will take SQuaRE ISO/IEC 25010 as a baseline for this new approach, because it is an improvement on ISO/IEC 9126-1 and now in force.

The first step was to debug the terminology, because we found that the standards used the same name to mean different things. Table 3.10 lists the characteristics and subcharacteristics of the SQuaRE in the left column and of the other models in the right-hand columns for comparison against SQuaRE, matching synonymous characteristics and synonymous subcharacteristics, such as the *understandability* subcharacteristic from ISO/IEC 9126-1, which has the same definition *appropriateness recognisability* in SQuaRE.

Table 3.10 Comparing usability characteristics against SQuaRE

Standards	SQuaRE-25010		ISO/IEC 9126-1		ISO 9241-11		QUIM consolidated model	
Characteristics	**Software product quality model**		**Internal and external quality model**		Usability		Usability	
	Operability	Appropriateness Recognisability	Usability	Understandability				
		Learnability		Learnability				Learnability
		Ease of use		Operability				
		Attractiveness		Attractiveness				
		Technical Accesibility						Accessibility
		Operability Compliance						
				Usability Compliance				
	System Quality-in-use model		**Quality in use**					
	Usability	Effectiveness		Effectiveness		Effectiveness		Effectiveness
		Efficiency		Productivity		Efficiency		Efficiency; Productivity
		Satisfaction	Likability	Satisfaction		Satisfaction		Satisfaction
			Pleasure					
			Comfort					
			Trust					
		Usability Compliance						
				Safety				Safety
								Trustfulness
								Universality
								Usefulness

From Table 3.10, we found some points worth considering in analyses in order to build the new model:

- *Usability compliance* from ISO/IEC 9126-1 has same definition as *usability compliance* (from SQuaRE), but they could not be matched because is classed as internal and external quality and the other as quality in use.
- Both *productivity* and *efficiency* from QUIM came under the same definition of efficiency (from SQuaRE). They are defined in QUIM as:
  - **Productivity** is the level of effectiveness achieved with respect to the resources (i.e., time to complete tasks, user efforts, materials or financial cost of usage) consumed by the users and the system.
  - **Effectiveness** is the capability of the software product to enable users to achieve specified tasks with accuracy and completeness.
  - **Efficiency** is the capability of the software product to enable users to expend appropriate amounts of resources in relation to the effectiveness achieved in a specified context of use.

  An efficient task is an accurate and complete task, which is the same as the definition of productivity. This is the reason why are lumped together.

- *Safety* is a subcharacteristic of both ISO/IEC 9126-1 and QUIM, whereas it is a characteristic of the SQuaRE quality-in-use model and classed at the same level as usability and flexibility. On this ground, it is omitted from our usability model.
- *Trustfulness* from QUIM is referred to in SQuaRE as *trust,* which is a sub-subcharacteristic of *satisfaction* and part of *usability* in the quality-in-use model.
- SQuaRE does not have a characteristic or a subcharacteristic that is equivalent to *usefulness* from QUIM. The most similar definition that we found was listed under the *functional appropriateness* subcharacteristic, which comes under the *functional suitability* characteristic. Here are the definitions:
  - **Usefulness (QUIM)** is whether a software product enables users to solve real problems in an acceptable way. Usefulness implies that a software product has practical utility, which in part reflects how closely the product supports the user's own task model. Usefulness obviously depends on the features and functionality offered by the software product. It also reflects the knowledge and skill level of the users while performing some task (i.e., not just the software product is considered).
  - **Functional appropriateness (SQuaRE)** is the degree to which the set of functions is suitable for specific tasks and user objectives.

  Usability does not imply utility, which is why we are not going to consider *usefulness* for the model. Utility depends of whether the requirements of the product are properly established from the beginning.

Table 3.11 shows the terminology that we finally chose to use to build the model that we propose.

Table 3.11 Choosing the terminology from the model comparison

Standards	SQuaRE-25010		ISO/IEC 9126-1		ISO 9241-11	QUIM Consolidated Model	Debugged new model	
Characteristics	**Software product quality model**		**Internal and external quality model**				**Software product quality model**	
	Operability	Appropriateness recognisability	Usability	Understandability			Operability	Understandability
		Learnability		Learnability		Learnability		Learnability
		Ease of use		Operability				Ease of use
		Attractiveness		Attractiveness				Attractiveness
		Technical accessibility				Accessibility		Technical accessibility
		Operability compliance						Operability compliance
						Universality		Universality
	System quality-in-use model		**Quality in use.**				**System quality-in-use model**	
	Usability	Effectiveness	Effectiveness		Effectiveness	Effectiveness	Usability	Effectiveness
		Efficiency	Productivity		Efficiency	Efficiency Productivity		Efficiency
		Satisfaction – Likability	Satisfaction		Satisfaction	Satisfaction		Likability
		Satisfaction – Pleasure						Pleasure
		Satisfaction – Comfort						Comfort
		Satisfaction – Trust						Trust
		Usability compliance						Usability Compliance

The second step was to establish canonical categories for the characteristics (see Table 3.12).

- ✓ The term HCI capability comes from human-computer interaction competence.
- ✓ Universal operability is related to universal usability, as reported in the website (Universal Usability), but was changed to universal operability to avoid conflicts with SQuaRE's definition of usability.

Table 3.12 Establishing canonical categories for the model

Characteristic	Canonical Characteristic	Subcharacteristic	Sub-subcharacteristic
Software product quality model			
Operability	HCI capability (human-computer interaction capability)	Understandability	
		Learnability	
		Ease of use	
		Attractiveness	
	Universal operability	Technical accessibility	
		Universality	
		Operability compliance	
System quality-in-use model			
Usability	Performance	Effectiveness	
		Efficiency	
	Satisfaction	Satisfaction	Likability
			Pleasure
			Comfort
			Trust
		Usability compliance	

Finally, as shown in Table 3.13, the following changes were made:

- The operability compliance, usability compliance and satisfaction were moved to a different level, because they were not members of a category.
- Universality was changed to cultural universality which better matches its definition.

Table 3.13 Resulting usability quality model

Usability Quality Model for Software Products		
Characteristic	**Subcharacteristic**	**Sub-subcharacteristic**
Software product quality model		
Operability	HCI capability	Understandability
		Learnability
		Ease of use
		Attractiveness
	Universal operability	Technical accessibility
		Cultural universality
	Operability compliance	
System quality-in-use model		
Usability	Performance	Effectiveness
		Efficiency
	Satisfaction	Likability
		Pleasure
		Comfort
		Trust
	Usability compliance	

3.4 Proposed Usability Model

Table 3.14 shows the model resulting from the analysis and debugging of the usability characteristics from different standards.

Table 3.14 Usability quality model for software products

Usability Quality Model for Software Products		
Characteristic	**Canonical Characteristic**	**Subcharacteristic**
Software product quality model		
Operability	HCI capability	Understandability
		Learnability
		Ease of use
		Attractiveness
	Universal operability	Technical accessibility
		Cultural universality
	Operability Compliance	
System quality-in-use model		
Usability	Performance	Effectiveness
		Efficiency
	Satisfaction	Likability
		Pleasure
		Comfort
		Trust
	Usability compliance	

This new approach describes a two-part model for software product quality:

- ✓ A *software product quality model*
- ✓ A *system quality-in-use model*.

The *software product quality model* explains the benefits of measuring *operability* in terms of *interaction ability*, *universal operability* and *operability compliance*, which are further divided into subcharacteristics. These subcharacteristics can be measured by internal and external metrics. Internal metrics are related to a non-executable software product during its development (such as a request for proposal, requirements definition, design specification,

inspections or source code). External metrics can be measured during the life cycle testing stages.

The *system quality-in-use model* is related to the user using the system in a realistic environment. This explains the benefits of measuring *usability* in terms of user *performance*, *satisfaction* and *usability compliance*.

The definitions of each component of this model follow. As mentioned above, our baseline is SQuaRE, from which we have borrowed some definitions.

- **Operability** is the degree to which the product has attributes that enable the user to interact with it, taking into consideration universal operability and following operability compliances.
  Users may include operators, end users and indirect users who are under the influence of or dependent on the use of the software.

  - **HCI capability** is the degree to which the product has attributes that enable it to be understood, learned, easy to use and attractive to the user[4], when used under specified conditions.
    - **Understandability** is the capability of the software product to enable the user to understand whether the software is suitable, and how it can be used for particular tasks and conditions of use.
    - **Learnability** is the degree to which the product enables users to learn its application.
    - **Ease of use** is the degree to which users find the product easy to operate and control.
    - **Attractiveness** is the degree to which the product is attractive to the user, which refers to attributes of software that increase user pleasure and satisfaction, such as the use of colour and the nature of the graphical design.
  - **Universal operability** is the degree to which the product can be used by people with or without disabilities (technical accessibility) and with a different cultural background (cultural universality).
    - **Technical accessibility** is the degree to which users with specified disabilities can operate the product. In order to provide equal access and equal opportunity to people with diverse abilities, such as older people, people in rural areas, people in developing countries and people with cognitive, neurological, physical, speech and visual disabilities.
    - **Cultural universality** is concerned with whether a software product accommodates a diversity of users with different cultural backgrounds (e.g., local culture is considered).

  - **Operability compliance** is the degree to which the product adheres to standards, conventions, style guides or regulations in laws and similar prescriptions relating to operability.

[4] **Users** may include operators, end users and indirect users who are under de influence of or dependent on the use of the software.

- **Usability** is the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use.

  - **Performance:**
    - **Effectiveness** is the accuracy and completeness with which users achieve specified goals.
    - **Efficiency** is the resources expended in relation to the accuracy and completeness with which users achieve goals.
  - **Satisfaction** is the degree to which users are satisfied in a specific context of use.
    - **Likability** is cognitive satisfaction.
    - **Pleasure** is emotional satisfaction.
    - **Comfort** is physical satisfaction.
    - **Trust** is satisfaction with security.
  - **Usability compliance** is the degree to which the product adheres to standards or conventions relating to usability.

References

- Bevan Nigel. (1999). Quality in use for all, In: User interfaces for all. Stephanidis, C (ed), Lawrence Erlbaum.
- Bevan Nigel. (2009). Extending Quality in Use to Provide a Framework for Usability Measurement, Proceedings of HCI International 2009.
- Bucur Ion I. (2006). On quality and measures in software engineering. Journal of Applied Quantitative Methods (JAQM), Vol.1 No.2.
- Colin Samuel, Maskoor Atif, Lanoix Arnaild, Souquières Jeanine, Hammad Ahmed, Dormoy Julien, Chouali Samir, Hufflen Jean-Michel, Kouchnarenko Olga, Mountassir Hassan, Lecomte Sylvain, Petit Dorian, Poirriez Vincent. (2008). A synthesis of existing approaches to specify non-functional properties, Project TACOS.
- Seffah Ahmed, Donyaee Mohammad, Kline Rex B., Padda Harkirat K. (2006). Usability measurement and metrics: A consolidated model. Software Qual J, Springer.

  IEEE Standards Status Report. [Online] Available from: <http://standards.ieee.org/cgi-bin/status?Computer%20Society/Software> [Accessed 15th January 2013]
- ISO Catalogue. [Online] Available from: <www.iso.org/iso/home/search.htm?qt=9241&published=on&active_tab=standards&sort_by=rel> [Accessed 15th January 2013]
- ISO 9241-11:1998, Ergonomics of Human System Interaction - Part 11: Guidance on Usability.
- ISO 9126-1:2001, Software engineering – Product quality – Part 1: Quality model.
- ISO 9126-2:2003, Software engineering – Product quality – Part 2: External metrics.
- ISO 9126-3:2003, Software engineering – Product quality – Part 3: Internal metrics.

- ISO 9126-4:2004, Software engineering – Product quality – Part 4: Quality in use metrics.
- ISO/IEC CD 25010.3:2009, Software Product Quality Requirements and Evaluation (SQuaRE) – Quality Models for Software Product Quality and System Quality in use.
- Universal Usability. . [Online] Available from: <http://universalusability.com/> [Accessed 24th January 2013]

4 Relationship between Security and Usability

This chapter addresses the process and results enacted to find a relationship between security and usability as quality factors. The results of this chapter have been accepted for publication by The 15th International Conference on Enterprise Information Systems ICEIS 2013, to be held in France in July 2013.

4.1 Introduction

The development of secure information systems has become critical in the last decade. As Ben-Asher et al. (2009) claim, the use of electronic transactions over the Internet and other networks, and the storage of an ever-increasing amount of sensitive data, is among the main factors behind this development.

Therefore, keeping information systems secure is by no means a simple and inexpensive task, as discussed by the Australian Government's Department of Defence (2012). Cyber security incidents can be costly, consuming financial and human resources. Examples of impacts are: service unavailability and loss of productivity, damage to the reputation of and customer's confidence in the targeted organization, lost or stolen information, loss of privacy, etc.

The Microsoft Security TechCenter (undated) reported a similar study, adding, however, more detailed information about costs. They consider direct and indirect costs, such as costs due to the loss of competitive edge as a result of the release of proprietary or sensitive information, legal costs, labour costs on the analysis of breaches, software reinstallation, and data recovery, costs of system downtime (for example, lost employee productivity, lost sales, replacement of hardware, software and so on).

On the other hand, Braz et al. (2007) pointed out that secure systems also need to be usable. Usability is, however, wrongly added on at the end of the life cycle development process because of the mistaken belief that security is related to the software system functionality and can be designed independently of usability which relates only to the UI component.

Mechanisms for ensuring security in information systems, such as authentication, sophisticated encryption algorithms and so on are only effective when they are configured and used correctly. As a result, security experts have identified users as being the weakest link in the security chain (Ben-Asher et al. 2009).

This is when we instinctively realize that there is a link between security and usability, and system users are the nexus between the two. We might even venture to say that usability goes hand in hand with security in the context of secure information systems.

The literature is replete with major debates concerning the relationship between security and usability. For instance, Cranor & Garfinkel (2005) refer to e-banking systems and some secure implementations that use the two-factor authentication

method. This method consists of a user-password and an automatically generated password generally sent to a mobile device for user authentication. This clearly detracts from system usability but is, at the same time, necessary in order to establish a level of security. As the authors state, “At first glance, the source of conflict might appear obvious: security usually aims to make operations harder to do, while usability aims to make operations easier. However, it’s more precise to say that security restricts access to operations that have undesirable results, whereas usability improves access to operations that have desirable results.”

From a software engineering (SE) point of view, security and usability are two independent quality factors that are specified in quality standards like (ISO/IEC 25010, 2011) (ISO/IEC 9126, 2001) and (ISO 9241-11, 1998).

The most recent software quality standard, ISO/IEC25010 also called SQuaRE (ISO/IEC 25010, 2011), defines security as “the degree of protection of information and data so that unauthorized persons or systems cannot read or modify them and authorized persons or system are not denied access to them”. On the other hand, usability is defined as “the extent to which a product can be used by specific users to achieve specific goals with effectiveness, efficiency and satisfaction in a specified context of use”. Additionally, the standards provide a set of subfactors for each of these quality attributes. The subfactors specified for security include confidentiality, integrity, non-repudiation, accountability, authenticity and security compliance, whereas, according to this standard, usability subfactors are effectiveness, efficiency and satisfaction.

Even though a thorough reading of the definitions of security and usability suggest that there is a relationship, the detailed link between the two concepts is today an open issue in the SE field. In this context, and bearing in mind that users are a key factor in secure systems, we believe that it would be important to find out of the details of the relationship between security and usability.

As far as we know, no other literature review has addressed this issue. On this ground, we conducted a systematic mapping study designed to summarize, analyse and understand what authors have been researching with respect to the relationship between security and usability over the last decade.

Our literature review focuses on security and usability as high-level quality factors. This is the prelude to more thorough-going research addressing the relationship between the characteristics and subcharacteristics of each of these two attributes.

To do this, the remainder of the paper is structured as follows. Section 4.2 outlines the research method that we have followed to perform the systematic mapping study. Section 4.3 reports the results, and Section 4.4 discusses our conclusions.

4.2 Research Method and Procedure

This section presents the process enacted to conduct our systematic mapping study of the literature related to security and usability. The guidelines provided by Petersen et al. (2008) were used to build this systematic map.

4.2.1 Research Question

As stated above, software quality models establish security and usability as quality factors. Accordingly, we formulated the following research question:

RQ. What type of relationship is there between security and usability?

This question intends to clarify the relationship between security and usability factors globally. Once we have the answer to this question, we will extend the mapping study to the characteristics and subcharacteristics of each factor.

4.2.2 Search Strategy

The search was run on several well-known databases, such as IEEE Xplore, ACM Digital Library and Inspec, as well as some individual journals and papers.
The publication year was set between 2002 and 2012 to limit the results to documents published within the last 10 years. Then, the titles and the abstracts of the identified articles were checked against set eligibility and relevance criteria (this criterion is explained in Section 4.2.3).
The literature was reviewed by a second researcher, and overlapping papers were removed.

4.2.3 Data Retrieval

Search strings were devised in order to collect information that can be used to answer the research question. The search strings were designed as follows:

X was composed of synonyms of or words possibly related to security in computer science, each linked by the “OR” operator.

X: {Security OR secure systems OR critical systems OR critical software OR security software constraints OR security software permissions}

Y was composed of synonyms of or words possibly related to usability in computer science, using each linked by the “OR” operator.

Y: {Usability OR operability OR user-centred design OR UCD OR human-computer interaction OR HCI OR user interface OR UI OR ergonomics}

Finally, the “AND” operator was added between X and Y to retrieve relevant literature related to security and usability (or respective synonyms) from the above data sources. Search string matching was confined to terms in the title and abstract of each publication.

4.2.4 Inclusion Process

The query strings devised in Section 4.2.3 were matched with the titles and abstracts of publications published in the last decade (2002-2012). As a result, the search process returned 120 papers for the three data sources. We read the abstract and introduction of these papers and discarded 55 papers as being

irrelevant. The other 65 were retained as relevant for the research. From the resulting 65 papers, 5 were duplicate publications, that is, multiple data sources returned the same papers. These duplicates were discarded, leaving 60 papers. Only 2 papers were unavailable from the electronic data sources, leaving 58 available full-text papers. We located another 17 papers in other sources, making a total of 75 papers. Of these, 33 were considered irrelevant and 42 papers were selected as being possibly useful for the research.

Finally, these 42 papers where analysed by reading the entire content in order to decide whether they were of any use for answering the research question. As a result, 25 papers were considered irrelevant, and only 17 papers were found directly related to security and usability.

4.3 Results

This section reports the results from the analysis of the seventeen papers detailed in the Appendix. First, we identify the different types of relationships identified by authors. Next, we classify the papers into research categories. Finally, we discuss the answers to the stated research question.

4.3.1 Taxonomy of Relationship Type

As already mentioned, the papers are detailed in the Appendix. From these papers, we established taxonomy for the different relationships identified by the authors. These relationships are listed and explained below.

Inverse relationship. Increasing or decreasing usability has the inverse or reciprocal effect on security and vice versa.
Direct relationship*.* Increasing or decreasing usability has the same effect on occur in security, and vice versa.
No relationship. The two factors are unrelated, so increasing or decreasing usability has no effect on security and vice versa.
One-way inverse relationship. There is an inverse relationship between the two factors, provided that the order is not changed.
Relative or dependent relationship. The relationship depends on some characteristic and could be direct in some cases and inverse in others.

Another point that researchers made is that the increases or decreases are not necessarily proportional. For example, increasing usability by 20% does not necessarily mean that security will decrease by the same percentage.

Table 4.1 summarizes the findings by author and type of relationship.

Table 4.1 Authors grouped by the type of relationship agreed

Legend:	
Inverse	-
Direct	+
Relative	*
One-way Inverse	/-
No Rela-tionship	o

Author	Minami et al. (2011)	Fidas et al. (2010)	Hahn et al. (2012)	Ben-Asher et al. (2009)	Braz &Robert (2006)	Kainda et al. (2010)	DeWitt & Kuljis (2006)	Roth et al. (2005)	Möckel (2011)	Epstein (2011)	Biddle et al. (2011)	Beckles et al. (2005)	Josang et al. (2007)	Mairiza & Zowghi (2010)	Prakash (2007)	Egyed & Grünbacher (2004)	Ferreira et al. (2009)
Security																	
Usability	-									+				*		/-	o

4.3.2 Papers Grouped by Research Category

We grouped papers according to the classification scheme proposed by Wieringa et al. (2006). The short description of each category follows.

Evaluation research is a technique or a solution implemented and evaluated in practice. In a *validation research*, techniques are novel, but still have not been implemented in practice, however, it probably tested in a laboratory environment. In a s*olution proposal*, a solution for a problem is proposed. In a *philosophical paper*, the field is structured in the form of taxonomy, outlining a new way of looking at existing things. In an *experience paper*, the personal experience of the author on what and how something happened in practice is reported. In an *opinion paper*, the personal opinion on a particular matter is expressed.

Papers were grouped as follows: six were classed as solution proposals, five, as evaluation research, and the experience paper, validation research and opinion paper categories each contained two papers. In the next section (4.3.3) are explained the papers and the research category assigned.

4.3.3 Relationships Grouped by Authors and Assignation of Research Category

In this section, we detail the different types of relationships between security and usability identified by the reviewed authors, and the type of research category assigned to the publication.

Inverse Relationship.

Table 4.1 shows that over half of the authors (nine out of seventeen) identified an inverse relationship between security and usability and vice versa. The reasons follow.

Minami et al. (2011).- This paper reports evaluation research addressing the trade-off between usability and security in the context of medical systems. The authors were tasked with adding an access control and encryption method to an existing system in order to protect patient security and privacy. They gathered feedback from health professionals in order to strike a balance between strict protection and usability.

Finally, they concluded that security compromises usability, but a good balance could be achieved. Their results are valuable for our research because they used a real system tested in a real environment, and this research used other subfactors, such as privacy and identification, which other papers often do not cover.

Fidas et al. (2010).- Based on their experience, the authors argue that system designers face contradicting requirements. On one side, stakeholders are willing to sacrifice user convenience in order to achieve system security, whereas, on the other, users are interested in system usability.

The argument suggests an inverse relationship, although the authors provide no evidence. The solution that they propose is a user-centric approach to the design of a secure and usable system. This does not provide a direct answer to our research question. However, their experience is considered to be relevant for our research.

Hahn et al. (2012).- This is a solution proposal paper analysing the security of many popular cloud services, such as CloudMe, Dropbox, HiDrive, etc. They analysed their vulnerabilities and listed possible attacks in an attempt to solve the discovered problems. In their analysis, they found that providers omitted email verification during registration, which is a security pitfall designed to avoid system usability problems.

From their analysis we can infer that increasing usability will decrease security and vice versa.

Ben-Asher et al. (2009).- This is a validation research paper reporting the construction of a controlled research environment that they called “microworld” to quantitatively evaluate and model the acceptability of security features as a function of the usability cost of their use, efficiency and threat severity. Their study is based on the behaviour of users overriding or ignoring security features to facilitate system use. This includes an alert system that warns about possible attacks, which, if not prevented, can cause losses of monetary earnings. Using the system they were able to manipulate the usability costs of using a security feature.

Their results showed that the percentage of usage was lower at a high security level, and the percentage of usage was higher at a low security level. This suggests that there is an inverse relationship between factors tested in a laboratory experiment.

Braz & Robert (2006).- This is an evaluation research paper, developing a comparative analysis of different authentication methods, such as passwords, PIN, proximity card, multifunctional card, public key, finger print and so on, and the perception of security that the user has. Results from their comparative analysis showed that methods that scored high on security were rated as having a significant level of usability issues. This applies to the listed authentication

methods, including multifunctional card, public key, and Kerberos and retina/iris techniques, which have a significant impact on the usability, and especially to retina analysis, because it requires more user cooperation. On the other hand, the authentication methods that scored low on security were rated as having a low level of usability issues. This applies to methods, such as passwords, PIN, and especially to voice, because it changes over the time, thereby decreasing security.

In other words, more security implies less usability, and less security implies more usability. They demonstrated an inverse relationship.

Kainda et al. (2010).- This is a solution proposal enacting a security-usability analysis process based on a threat model. The authors highlight a difference between standard security threat models and the HCISec (HCI over information security systems) model, because malicious attackers may or may not be legitimate users in standard models, whereas HCISec focuses on legitimate user mistakes that may compromise the system. Legitimate users make mistakes that breach system security because threat scenarios are more usable than secure scenarios. The idea is to make secure scenarios more usable to reduce the risk of users making mistakes.

We infer from the view that the authors take that the relationship between factors is inverse.

DeWitt & Kuljis (2006).- This is an evaluation research paper conducting a usability study on Polaris, which was built to make the Windows operating system safer from viruses and malicious code, but highly usable as well as secure. The study used a laboratory test during which users were asked to perform tasks that included the use of security. They measured learnability and usability based on standard ISO 9241-11, which defines usability as comprising effectiveness, efficiency, and satisfaction. With respect to security, they include tasks such as browsing an Internet banking website, checking emails and following hyperlinks to try out attached files.

They found that participants were willing to compromise security. They reasoned this behaviour by declaring that the speed and ease with which tasks were completed is more important than the protection of their files. This means that people favour ease of use over security.

Roth et al. (2005).- This is an evaluation research paper conducting an experiment where security mechanisms were applied to protect mails sent by non-commercial users in order to find the correct trade-off between usability and security. The authors researched what the optimum trade-off should be and how security benefits can be maximized with minimum damage to usability. In the case of usability, they improved the user interface by decreasing the number of interactions, and letting users handle concepts with which they were familiar, like, for example, “mail” rather than “keys”. They aimed to provide a familiar context and mental model representation. In the case of security, they tried to protect the email communication from external attack and assure the data integrity.

This experiment was conducted on a particular security field, known as Attack/Harm Detection and Integrity, where the user interacts as little as possible with the issues related to security. Although this is a particular case at the subfactor

level, it is a good result as far as we are concerned because few researchers have considered these security issues.

Möckel (2011).- This is a solution proposal paper aiming to build an evaluation framework to align usability and security in the context of e-banking systems. We are primarily concerned with one of its research questions, namely, What is the relationship between security criteria influencing mitigation quality and usability criteria?

Möckel intimates that the relation is inverse, stating that "Most works have focused on the balance between security and usability in regard to authentication methods, often in a comparative fashion or on individual solutions. Author Murdoch discusses the problem of 'usability optimization' with a negative effect on overall system security…" Unfortunately, this research is incomplete and, consequently, we class it as an opinion paper.

Direct Relationship.

Four out of the seventeen authors identified a direct relationship between security and usability and vice versa on the following grounds.

Epstein (2011).- This is an experience paper establishing that it is possible to align usability and security as part of the software development cycle and not as an add-on at the end.

We interpret this as meaning that there is direct relationship between the two factors: "On occasions, security and usability are perceived as being at odds… However, this is a false dichotomy –usability also requires preventing inappropriate actions…"

Biddle et al. (2011).- This is an opinion paper expressing the viewpoint that usability and security are directly related because when one is negatively affected the other will be negatively affected as well: "Usability and security are not simply inversely related, most especially because faults in one lead to faults in the other..." The authors based their opinion on their experience.

Beckles et al. (2005).- This is a solution proposal paper describing two proposals to improve grid security usability.

Grid security is based on public key infrastructure (PKI), but PKI implementations have unfortunately suffered, from serious usability issues in terms of end-user acquisition and management of credentials. The specific usability issues that they found were credential acquisition, configuration complexity, mobility, user management of credentials and revocation. The approach of the first solution, named PKIBoot, is for everything to be automatically configured via user-password authentication, i.e., the client has nothing to do with settings. According to the second solution, named GridLogon, users log in for a service to access their entire security configuration.

Regarding the relationship between grid usability and security their results suggest that improvements in usability (in the area of credential management) are required if the security of these environments is to be maintained.

Josang et al. (2007).- This is a solution proposal paper reporting a set of security usability vulnerabilities that can be used to assess the risk to secure systems. The solution is designed at the usability and security factor level, but the example only shows how to apply the solution at the level of one subfactor, namely, authentication for web security solutions.

The authors analyse whether upgrading usability improves security. This proposal would benefit from evaluation in practice, but the examples and the way in which the authors illustrated their solution is quite suitable for our purpose.

Relative Relationship.

Mairiza & Zowghi (2010).- This is a solution proposal paper cataloguing conflicts between non-functional requirements (NFR) like accuracy, usability, availability, reliability, security, etc. The conflicts were categorized as absolute conflict, relative conflict and never conflict. Their results establish that conflict between usability and security (and vice versa) is relative, because they conflicted sometimes but not always.

Prakash (2007).- This is a validation research paper reporting two studies of vulnerabilities at the user level. The first study was conducted on a group of computer science graduate students and some faculty (non-regular users). It examined the vulnerability of users to man-in-the-middle attacks on SSH. The second study analysed over 700 bank web sites for a range of vulnerabilities that resulted from poor website design decisions from a security perspective.

For the first experiment, the results indicate a direct relationship: there are widespread security problems at the end-user level in systems that attempt to deploy security protocols to secure user interactions. Advanced users bypassed the SSH security warning to log on to the server, perceiving it as an obstacle because SSH does not indicate what to do to solve the problem save contacting the administrator. This means that usability was not good enough causing security problems. From this we infer that poor usability compromises security. For the second experiment, results show an inverse relationship: setting the credential inputs on an insecure page (even when the data were sent to the server in a secure form) in order to increase the usability caused a security problem. From this we infer that good usability compromises security.

One-Way Inverse Relationship.

Egyed & Grünbacher (2004).- This is a solution proposal paper consolidating a matrix of potential conflicts and cooperation between quality attributes from a wide range of literature and ISO 9126, such as functionality, efficiency, usability,

accuracy, security and so on. This model takes into account that attributes might be indifferent to one another, cooperative or conflicting.

They found that usability is indifferent with respect to security. Inversely, however, security enters into conflict with usability. In other words, improving usability does not affect system security (there is no relationship). In the other case, however, security is inversely related to usability, because, if it increases, system usability will decrease.

No relationship.

Ferreira et al. (2009).- This is an evaluation research paper consolidating a model of patterns to align usability and security. Factor patterns, such as explicit user audit, complete delete, email-based identification and authentication and so on (previously created by Garfinkel et al.). The authors validate these patterns in an experiment with computer experts, who managed to apply 61.67% of these patterns in software, concluding that they are a useful guide for developers.

These authors establish that there is no conflict between security and usability, stating that "Some authors argue that it can be complicated to build systems with both security and usability, but the reality is that there is no real conflict between these two properties". Following these guidelines, they were able to build secure software with a trade-off and without conflicts between security and usability.

4.3.4 Discussion

Figure 4.1 shows the five types of relationship between security and usability on the horizontal axis, whereas the six research categories appear on the vertical axis.

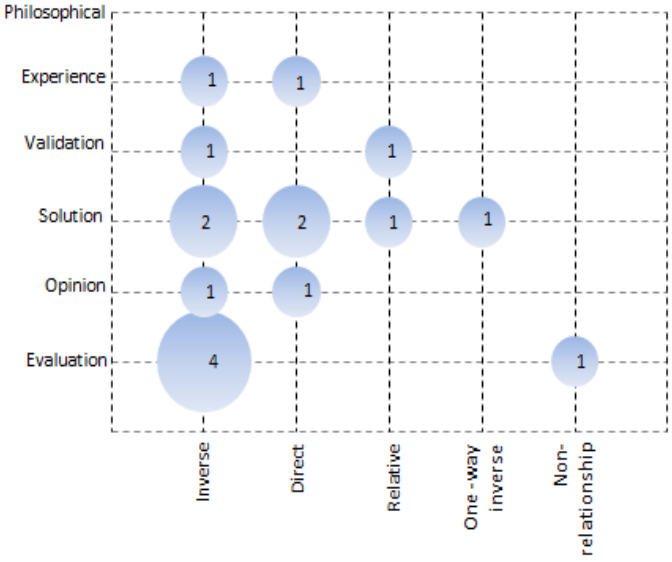

Figure 4.1 Distribution of papers by category and relationship

Clearly, most of the authors (nine) agree that security and usability are inversely related. The next group (four) agreed that there is a direct relationship. Only two authors found a relative relationship. The one-way inverse relationship and non-relationship are supported by only one author, respectively.

We consider evaluation research to be the best method for gathering evidence, because it reports solutions that have been implemented and evaluated in practice in a real environment.

As Figure 4.1 shows, four out of nine papers report evaluation research that found that there was an inverse relationship, which is the relationship upon which most authors agreed.

This evaluation research was conducted in a specific setting, including access control and encryption, authentication methods, attack detection in an operating system, and attack detection over emails.

From these results, we can conclude that the inverse relationship is the most reliable response to our research question. However, we also found some evidence of contrary findings.

There is one prominent case. Prakash (2007) reports validation research showing a relative relationship between security and usability. He found a direct relationship in one experiment and an inverse relationship in another experiment. The first experiment involved computer science students using Secure Shell (SSH), which is a cryptographic network protocol. The second experiment was about vulnerabilities produced by poor user interface decisions.

Comparing these two results, the relationship appears in our opinion to be relative, that is, it depends on the subfactor of security and/or usability to be measured. Therefore, more research is required at the subfactor level, as it is far from straightforward to define a relationship from a global perspective between security and usability without analysing each of their subfactors.

Another point is that the relationship probably depends of the application type. However, more research is needed to formally assure this issue.

There is not much evidence to defend the other relationships (direct, one-way inverse and no relationship). For example, one paper that found a direct relationship is based on experience and the other exemplifies the author's opinion. This means that there is no real quantified evidence. Finally, the other two are solution proposals which required evaluation in a real environment.

Only one author proposes a one-way inverse relationship. He claims that security changes alter usability, but the opposite does not apply. From this we infer that this relationship is probably a particular case of a relative relationship.

One paper suggests that security and usability are not related. In this case, the authors claim that there is no conflict between these factors. However, we think that their solution targets the alignment of these two factors. In this case, striking a balance between security and usability in order to improve these two factors does not necessarily mean that there is no conflict.

A significant number of papers examine the trade-off between security and usability to strike the correct balance between the two factors. This balance is important, because users are unlikely to use a 100% secure system if this significantly compromises usability, for example, by 50%.

Other authors claim that there is a relationship between security and usability, although they do not provide explicit details about the connections. This is the case

of Cranor & Garfinkel (2005), who have identified users as the weakest link in the security chain. This is a clear sign of there being a relationship, because it is the users that operate the systems and, at the same time, handle sensitive information like passwords, bank information, and so forth.

4.3.5 Threats to Validity

Until we find other systematic reviews focusing on the taxonomy of the relationship between security and usability, we will not be able to validate our study externally. Regarding internal validity, the author and the supervisors were involved in this systematic mapping study. We discussed and agreed on the procedure and considered activities to counteract the effect of researcher bias. On the other hand, we used general terms and placed no constraints on the search strings in order to achieve better coverage as well as high accuracy. We selected three of the most important electronic data sources to which we had access, and added other external sources. The chosen time-frame was intended to include the last decade of research.

During the exclusion process, we were particularly careful not to discard any potentially interesting paper. For this reason, we also included papers whose abstract or title was not completely clear with respect to our research question for further reading. It was also rather difficult to distinguish the research focus of some papers. So, even though we agree on the result of this process, replicating authors might categorize the studies differently.

The papers do not directly answer the research question, and we had to identify the response by analysing and inferring what authors had found in their research. However, we negotiated this process.

4.4 Conclusions

Our systematic mapping found five types of relationships between security and usability addressed by authors during the last decade. These types are: inverse, direct, relative, one-way inverse and no relationship.

The inverse relationship is the most often mentioned connection, followed by the direct, and relative relationships, where the least supported associations are one-way inverse and no relationships.

We based our results on the type of research conducted by the authors, where more weight is attached to evaluation research because results are more reliable.

A significant number of authors agreeing on an inverse relationship conducted evaluation research; however, another author who ran two different experiments found a relative relationship. Therefore, it is also necessary to analyse whether application type influences the results.

There is less evidence to support the other relationships (direct, one-way inverse and no relationship).

In sum, the literature regarding the relationship between security and usability published over the last decade is not unanimous. This point required further empirical research to study this relationship at the level of characteristics and subcharacteristics of both quality factors. This is the next step in our research.

References

- Ben-asher, N., Meyer, J., Parmet, Y., Moeller, S., Englert, R. (2009) An Experimental System for Studying the Tradeoff between Usability and Security. International Conference on Availability, Reliability and Security.

- Australian Government's Department of Defence: Preparing for and Responding to Cyber Security Incidents. (2012). [Online] Available from: http://www.dsd.gov.au/publications/csocprotect/preparing_for_cyber_incidents.htm [Accessed 5th April 2013]

- The Microsoft Security TechCenter: Responding to IT Security Incidents. (no date). [Online] Available from: http://technet.microsoft.com/en-us/library/cc875825.aspx [Accessed 5th April 2013]

- Braz, C., Seffah, A., M'Raihi, D. (2007). Designing a Trade-off between Usability and Security: A Metrics Based-Model. Springer volume 4663, 114-126

- Cranor, L., Garfinkel, S. (2005). Security and Usability Designing Secure Systems that People Can Use. INTERNATIONAL Journal of Computers and Communications Issue 1, O'Reilly Media.

- ISO/IEC, 2011, ISO/IEC 25010, Software Product Quality Requirements and Evaluation (SQuaRE) – Quality Models for Software Product Quality and System Quality in use. International Standard. Switzerland.

- ISO 9126-1, 2001, Software engineering – Product quality – Part 1: Quality model. International Standard. Switzerland.

- ISO 9241-11, 1998, Ergonomics of Human System Interaction - Part 11: Guidance on Usability. International Standard. Switzerland.

- Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M. (2008). Systematic Mapping Studies in Software Engineering. 12th International Conference on Evaluation and Assessment in Software Engineering.

- Wieringa, R., Maiden, N., Mead, N., Rolland, C. (2006). Requirements Engineering Paper Classification and Evaluation Criteria: a Proposal and a Discussion. Journal of Requirements Eng.

Appendix

A detailed list of the literature reviewed in this article follows:

- Beckles, B., Welch, V., Basney, J. (2005). Mechanisms for increasing the usability of grid security. Int. J. Human-Computer Studies 63, 74–101
- Ben-asher, N., Meyer, J., Parmet, Y., Moeller, S., Englert, R. (2009). Security and Usability Research Using a Microworld Environment. Deutsche Telekom AG as part of the Telekom Laboratories.
- Biddle, R., Brown, J., Chiasson, S., Martin, A. (2011). Security Dialectics and Agile Software Development. Position Paper for the 1st Software and Usable Security Aligned for Good Engineering (SAUSAGE) Workshop.
- Braz, C., Robert, JM. (2006). Security and Usability: The Case of the User Authentication Methods. International Journal of Human-Computer Studies.
- Dewitt, A., Kuljis J. (2006). Aligning Usability and Security: A Usability Study of Polaris. SOUPS '06 Proceedings of the second symposium on Usable privacy and security.
- Egyed, A., Grünbacher, P. (2004). Identifying Requirements Conflicts and Cooperation: How Quality Attributes and Automated Traceability Can Help. Focus persistent software attributes.
- Epstein, J. (2011). Integrating Security & Usability Into the Software Development Lifecycle. Position Paper for the 1st Software and Usable Security Aligned for Good Engineering (SAUSAGE) Workshop.
- Ferreira, A., Rusu, C.: Roncagliolo, S. (2009). Usability and Security Patterns. Second International Conferences on Advances in Computer-Human Interactions.
- Fidas, C., Voyiatzis, A., Avouris, N. (2010). When Security Meets Usability: A User-Centric Approach on a Crossroads Priority Problem. 14th Panhellenic Conference on Informatics.,
- Hahn, T., Kunz, T., Scheneider, M., Sven, V., Moeller, S., Englert, R. (2012). Vulnerabilities through Usability Pitfalls in Cloud Services: Security Problems due to Unverified Email Addresses. IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications.
- Jøsang, A., AlFayyadh, B., Grandison, T., AlZomai, M., McNamara, J. (2007). Security Usability Principles for Vulnerability Analysis and Risk Assessment, 23rd Annual Computer Security Applications Conference.
- Kainda, R., Flechais, I., Roscoe, A. (2010). Security and Usability: Analysis and Evaluation, International Conference on Availability, Reliability and Security.
- Mairiza, D., Zowghi, D. (2010). Constructing a Catalogue of conflicts among non-functional Requirements, 5th International Conference, ENASE, Athens, Greece.

- Minami, M., Suzaki, K., Okumura, T. (2011). Security considered harmful A case study of trade-off between security and usability. The 8th Annual IEEE Consumer Communications and Networking Conference - Work in Progress (Short Papers).
- Möckel, C. (2011). Usability and Security in EU E-Banking Systems towards an Integrated Evaluation Framework. IEEE/IPSJ International Symposium on Applications and the Internet.
- Roth, V., Straub, T., Richter, K. (2005). Security and usability engineering with particular attention to electronic mail. International Journal of Human-Computer Studies.
- Prakash, A. (2007). Security in Practice – Security-Usability Chasm. Information Systems Security Lecture Notes in Computer Science Volume 4812, 1-9

5 Relationship between Authentication and User Efficiency

This chapter addresses the relationship between authentication and user efficiency at the quality subfactor level. The research follows.

5.1 Introduction

Authentication mechanisms represent the first breakable point in the security chain of secure information systems, because they are the main way in.

The definition provided by Helkala (2012) is that “authentication is a visible element of information protection and its essence is that users prove that they are who they claim to be”. Three processes have been identified during authentication. They are identification, authorization and access control.

Identification takes place first. Jali et al. (2011) give a description of this process: “User authentication can be explained as a process of proving user identity for a particular service or system that they wish to use”.

Authorization is the second step in the authentication process. Kroeze and Oliver (2012) claim that this process involves “… knowing who a user of a system is and denying access to unauthenticated users”, the phrase ‘denying access’ goes further than mere identification, that is, it is the process whereby the identified user is authorized to access the system.

Access control is the last step in the authentication process, whereby identified and authorized users are assigned resources to perform certain tasks, like modify, delete, or create.

Nowadays, there are many techniques for user authentication, ranging from traditional username/password to the use of biometrics, as stated by Jali et al. (2011). Examples of these techniques are:

- Text-based credentials, such as textual passwords, PINs, challenge questions (Helkala, 2012)
- Out-of-band authentication (OOB), like one-time passcode (OTP), which are codes generated to be used once then expire; these codes are sent using another communication channel such as cell phone communication networks (Goyal et al., 2013)
- Image-based authentication, including click-based graphical password, choice-based graphical password and draw-based graphical password, (Meng, 2012), as well as CAPTCHAs
- Biometrics, which can be divided into three categories: *biological*, like blood, odour and saliva; *behavioural*, like signature, keyboard typing, gait; and *morphological*, like iris, fingerprint, face, hand geometry, as pointed out by Hentati et al. (2012)

According to Eljetlawi (2010), a text-based password is a very common and still widely used authentication method these days. However, these common passwords have been found to have some drawbacks; for example, they may be stolen, forgotten or weak, etc. Complex text-based passwords use a long unfamiliar character string that increases the memory load (sometimes causing the user to write down the password, which is considered as a security drawback), which attackers find it tough to guess. The login success rate relates to the length of the password. As a result, the users may forget or mistype their passwords (Han et al., 2011). Easy text-based passwords use a familiar password, like family names, pet names or even the word “password”, which was reported by The Daily Mail to be the top common character string used as a password in 2012. They are therefore susceptible to brute force attack (which is an attempt to randomly guess the password) or shoulder surfing.

Graphical-based passwords were created to solve the deficiencies caused by text-based passwords. This method involves recognizing images rather than recalling alphanumeric passwords as stated by Chowdhury & Poet (2011). However, they also are susceptible to attacks like hot-spot attacks (where attackers create a dictionary of popular spots for images or points).

Biometric authentication is another method, whereby the user is continuously identified (Mock et al., 2012) without even noticing, and the session can be closed at any time if an intruder is detected.

Eljetlawil (2010) suggests a set of features that a usable password should have. It should be easy to use, easy to memorize, easy to create, easy to learn, whereas the process should be pleasant for the user (satisfaction) and the login time should be reasonable. These characteristics are usability attributes (ISO/IEC9126, 2001). So, there is a potential relationship between usability and the authentication process.

According to the latest quality standard SQuaRE (ISO/IEC25010, 2011); product usability can be estimated by measuring its subfactors: effectiveness, efficiency and satisfaction.

Efficiency is defined by SQuaRE as: “The resources expended in relation to the accuracy and completeness with which users achieve goals”. Seffah et al. (2006) claim that efficiency is determined by the estimated costs (e.g., total time) of executing user procedures.

On this account, we consider the time spent by the user on the authentication process (including user errors and user help time) as the resource to measure the efficiency in this research. In this context, the purpose of this research is to find out the relationship between authentication (as part of security) and user login efficiency (as part of usability).

As far as we know, no other literature review has addressed this issue. On this ground, we conducted a systematic mapping study designed to summarize, analyse and understand what authors have been researching with respect to the relationship between authentication and user login efficiency over the last decade.

To do this, the remainder of the paper is structured as follows. Section 5.2 outlines the research method that we have followed to perform the systematic mapping study. Section 5.3 reports the results, and Section 5.4 discusses our conclusions.

5.2 Research Method and Procedure

This section presents the process enacted to conduct our systematic mapping study of the literature related to authentication and user login efficiency. The guidelines provided by Petersen et al. (2008) were used to build this systematic map.

5.2.1 Research Question

The research question formulated for this paper intends to clarify the relationship between authentication (as a security subfactor) and user efficiency (as a usability subfactor).

> **RQ.** What type of relationship is there between authentication and user login efficiency?

As stated earlier, efficiency has to be measured using resources. In this research we focused on authentication resource time, also known as login time.

5.2.2 Search Strategy

The search was run on several well-known databases, such as IEEE Xplore, ACM Digital Library and Inspec, as well as some individual journals and papers. The publication year was set between 2003 and 2013 to limit the results to documents published within the last 10 years.

The titles and the abstracts of the identified articles were checked against set eligibility and relevance criteria (explained later in Section 5.2.4).

The literature was reviewed by a second researcher, and overlapping papers were removed.

5.2.3 Electronic Data Sources (EDS)

The following electronic data sources were used:

1) *IEEE Xplore® digital library*,
   available at: http://ieeexplore.ieee.org
   IEEE Xplore® access IEEE journals, transactions, letters, magazines and conference proceedings, IET journals and conference proceedings, IEEE standards and IEEE educational courses.
2) *ACM Digital Library (DL),*
   available at: http://dl.acm.org/
   ACM Digital Library (DL) provides access to an up-to-date collection of full-text articles and bibliographic records covering the fields of computing and information technology and including all the ACM's journals, conference proceedings, magazines and newsletters.
3) *Inspec,*
   available at: http://www.accesowok.fecyt.es/

Accessed from the Web of Knowledge (WOK), a Thomson Reuters web-based platform, Inspec consists of a large collection of bibliographic databases, quotations and references to scientific publications in any scientific and technological, humanistic and sociological branch of knowledge.

5.2.4 Data Retrieval

Search strings were devised in order to collect information that can be used to answer the research question. The search strings were designed as follows:

X was composed of synonyms of or words possibly related to authentication in computer science, each linked by the “OR” operator.

X: {Authentication OR authorization}

Y was composed of synonyms of or words possibly related to efficiency of usability in computer science, each linked by the “OR” operator.

Y: {user efficiency OR login time OR access time}

Finally, the “AND” operator was added between X and Y to retrieve relevant literature related to security and usability (or respective synonyms) from the above data sources.

Search string matching was confined to terms in the title, keywords and abstract of each publication.

5.2.5 Inclusion Process

The query strings devised in Section 5.2.4 were matched with the titles and abstracts of publications published in the last 10 years (2003-2013). As a result, the search process returned 68 papers for the three data sources.

We read the abstract and introduction of these papers and discarded 14 as being irrelevant, because the content was related to networking protocols. The other 54 were retained as relevant for the research.

Of the resulting 54 papers, 11 were duplicate publications, that is, multiple data sources returned the same papers. These duplicates were discarded, leaving 43 papers.

We located another 6 papers from other sources, making a total of 49.

Finally, these 49 papers were analysed in order to decide whether they were of any use for answering the research question. As a result, 40 were considered irrelevant, and only 9 papers were found to be directly related to authentication and user login efficiency. This process is illustrated in Figure 5.1.

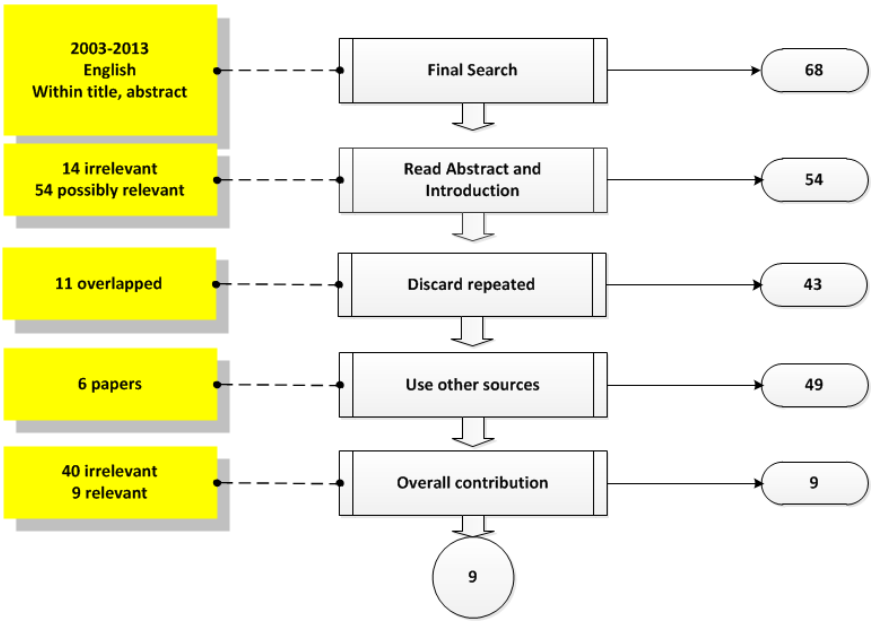

Figure 5.1 Inclusion process

Table 5.1 shows the overlap matrix. The number in each cell indicates how many papers overlap. A total of eleven papers overlapped.

Table 5.1 Overlap matrix

	IEEE	ACM	Inspec
IEEE		0	11
ACM			0
Inspec			

5.3 Results

This section reports the collected data, our analysis and, finally, discusses the answers to the stated research question.

5.3.1 Papers Grouped by Research Category

As already mentioned, 49 papers were analysed. Of these, 40 were considered as irrelevant and 9 relevant for the research. The papers are detailed in the appendix.

The research type is an important indicator of how reliable the results of each paper are. Research type indicates whether the paper is an opinion, a new solution proposal, a test in a controlled environment or a test in a real environment. On this

ground, we grouped papers according to the classification scheme proposed by Wieringa et al. (2006).The description of each category follows.

Evaluation research is a technique or a solution implemented and evaluated in practice. *Validation research* reports novel techniques that have not yet been implemented in practice, but have probably been tested in a laboratory environment. A *solution proposal* proposes a solution for a problem. A *philosophical paper* structures the field in the form of taxonomy, outlining a new way of looking at existing things. An *experience paper* reports the personal experience of the author on what and how something happened in practice. An *opinion paper* expresses the personal opinion on a particular matter.

Figure 5.2 shows the number of relevant papers during each publication year and their respective research category. Papers were grouped as follows: four were classed as solution proposals, four as evaluation research, and one as validation research. Figure 5.2 contains no papers belonging to philosophical, experience or opinion categories because we searched papers containing measurements of login time.

The research reported in these papers dates from 2009 until the present, probably because there has been more interest in the use complex systems with authentication restrictions in the last few years.

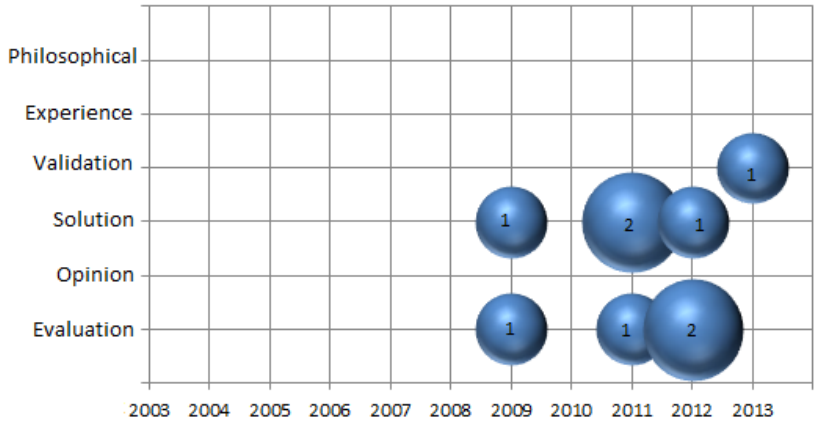

Figure 5.2 Categorization of papers by research type

5.3.2 Login Efficiency during the Authentication Process

Below, we analyse the methods used during the authentication process, ranging from text-based, graphical-based, biometrics and hybrid systems.

Note that we choose papers that report measurements of the login time spent on the authentication process.

Results are summarized in Table 5.2.

Perkovic et al. (2009)

This is an evaluation research paper, where authors create a PIN-based scheme for authentication called SSSL (Shoulder Surfing Safe Login).

In SSSL, the user knows the PIN number; the system gives challenge values which are different in each login. The procedure consists of locating the immediate (one-hop) neighbourhood between the PIN and the challenge values, and then the user presses the arrow that represents the movement to that number.

There were 15 volunteers for the experiment (half of the original 30 recruits dropped out, probably because they were given no incentives). Finally, the SSSL average login time was around 8 seconds.

Ma &Feng (2011)

This is an evaluation research paper, conducting a study to examine the usability of three authentication methods by measuring login time, login failure rate, cognitive load ratings (mental, frustration, temporal, etc.) and so on. The evaluated methods are: text-password, mnemonic password, and graphical password.

Text-password: users were required to select a strong password, for example, with a length of 6 to 20 characters, mixture of upper and lower cases and numbers; and the possibility of using special symbols.

Mnemonic password: they used a text password based on a particular phrase.

Graphical password: they used recognition-based graphical password.

There were 26 participants in the experiment, 9 for text, 10 for mnemonic, and 7 for graphical-based passwords.

The results for the average of login time were approximately 10 seconds for text and mnemonic, and approximately 25 seconds for graphical passwords, all this measurements during the second visit (the first visit was to create the account and login once). The most time-consuming option was the graphical password, whereas there was no significant difference between text and mnemonic.

Jali et al. (2011)

This is a solution proposal paper, combining two graphical password methods for better security.

The methods are click-based and choice-based graphical passwords. Click-based passwords require users to click certain points of the images from clicking areas, established during account creation. Choice-based passwords require users to select images from a group, previously selected during account creation, in a specific order.

There were 30 participants in the experiment; 20 were male and 10 female. The mean time for authentication (without account creation) was 25 seconds and the longest was 50 seconds.

Gao et al. (2009)

This is a solution proposal paper, where authors created a new solution by introducing a modification to the choice-based graphical password.

The method required users to select icons whose background has a predefined colour throughout the password, so users merely have to pay attention to the icons with this colour. Shoulder surfing attack is avoided by replacing the selected icon during login time with a wildcard.

There were 30 volunteers in the experiment. The results show that it took the user approximately 7 seconds to login into the system in the second login and approximately four seconds during successive logins (from 5 to 10).

Although this solution is slower than text-based schemes, 85% of participants think that this is acceptable.

Zangooei et al. (2012)

This is a solution proposal paper, proposing and evaluating a new hybrid graphical-based authentication scheme.

The hybrid is composed of a recognition-based and recall-based password to solve the conflicts of the separate solutions.

In the first part, the user has to recognize photos, which are then replaced by random alphanumeric characters that have to be typed by the user.

There were 30 participants in the experiment, 10 females and 20 males.

They measured the average login time, which turned out to be 8 seconds in the first test and 7 seconds in the second one.

Meng (2012)

This is an evaluation research paper, improving usability by using graphical passwords instead of text-based passwords.

Meng compared two methods: DAS and CD-GPS. DAS (draw a secret) users draw their own passwords on a 2D grid. The user has to reproduce the same sequence in the authentication. CD-GPS (click-draw based graphical password) consists of image selection and secret drawing.

Time login results for CD-GPS are 13.7 seconds and 25.7 seconds for DAS.

There were 42 participants in the experiment; 23 were female and 19 male.

Liu et al. (2011)

This is a solution proposal paper, where authors came up with a scheme called CBFG, which stands for click buttons according to figures in grids.

This is another graphical password scheme. Results show that the mean login time was 21.4 seconds, but this time dropped after a prolonged period of use.

There were a total of 24 participants, 14 males and 10 females.

Sayed et al. (2013)

This is a validation research paper, where authors built a new framework for fast authentication using mouse dynamics biometrics.

During the enrolment process the user draws a set of gestures using a mouse, which later are used for identification purposes. This process has to be repeated several times, because enough data has to be captured for user recognition, and the prototype has some limitations on the maximum number of gestures that could be used.

In the experiment, the verification time for the maximum combination of gestures (set at four) allowed by the prototype is five seconds (5 s).

A total of 39 volunteers participated in the experiment.

Mock et al. (2012)

This is an evaluation research paper, using a commercial eye-tracker to evaluate a continuous real-time authentication system via iris recognition.

The results report the percentage of correct authentication, 11% of EER equal error rate. The authors use an algorithm that can reduce processing time to about 2 seconds per sample.

There were 37 participants in the experiment.

As stated above, Table 5.2 shows a summary of the login time results by authentication scheme.

Table 5.2 Login time results for different schemes

Login Time During Authentication												
Author	Perkovic et al.	Ma & Feng			Jali et al.	Gao et al.	Zangooei et al.	Meng.		Liu et al.	Sayet et al.	Mock et al.
Year	2009	2011			2011	2009	2012	2012		2011	2013	2012
Scheme	Shoulder Surfing Safe Login (SSSL)	Text	Mnemonic	Graphical (Recognition - Based)	Graphical (Click-based & Choice-based)	Color Login Graphical password	Graphical based Hybrid Solution (recogniton & recall)	Click & Draw - based Graphical Password (CD-GPS)	Draw a Secret (DAS) Graphical	Click Buttons according to Figures in Grids (CBFG)	Mouse Dynamics Biometrics (Draw Gestures)	Eye Tracker
Time (seconds)	8	10	10	25	25	7	7	13.7	25.7	21.4	26.9	2
Volunteers	15	26			30	30	30	42		24	39	37
Male	-	20			20	-	20	19		14	-	-
Female	-	6			10	-	10	23		10	-	-

5.3.3 Discussion

The schemes addressed by the selected papers are of various types (text-based, graphical-based and biometrics), and combinations of the above, as summarized in Table 5.2.

As we were looking for a behavioural pattern, we decided to set a security level. Gao et al. (2009) claim that “Graphical passwords are believed to be more secure than traditional textual passwords… However, biometric-based passwords are believed to provide the highest level of security”.

In this context, we set the security level as “low” for text-based, “medium” for graphical-based, and “high” for biometrics passwords, as shown in Table 5.3. This information was used to plot a chart, shown in Figure 5.3.

Table 5.3 Security levels assignation (according to Gao et al., 2009)

Scheme	Shoulder Surfing Safe Login (SSSL)	Text	Mnemonic	Graphical (Recog nition - Based)	Graphical (Click-based & Choice-based)	Color Login Graphical password	Graphical based Hybrid Solution (recogniton & recall)	Click & Draw - based Graphical Password (CD-GPS)	Draw a Secret (DAS) Graphi cal	Click Buttons according to Figures in Grids (CBFG)	Mouse Dynamics Biometrics (Draw Gestures)	Eye Tracker
Login Time	8	10	10	25	25	7	7	13.7	25.7	21.4	26.9	2
Security Level	Low			Medium							High	

The chart illustrated in Figure 5.3 shows that, except for text-based passwords, the login time for the same security level can be high and low. For example, the login time for biometrics catalogued as having a “high security level” is low for the eye tracker method and high for mouse dynamics. The same applies for the medium security level. Summarizing, efficiency is affected by the authentication method and not by the level of security used during authentication.

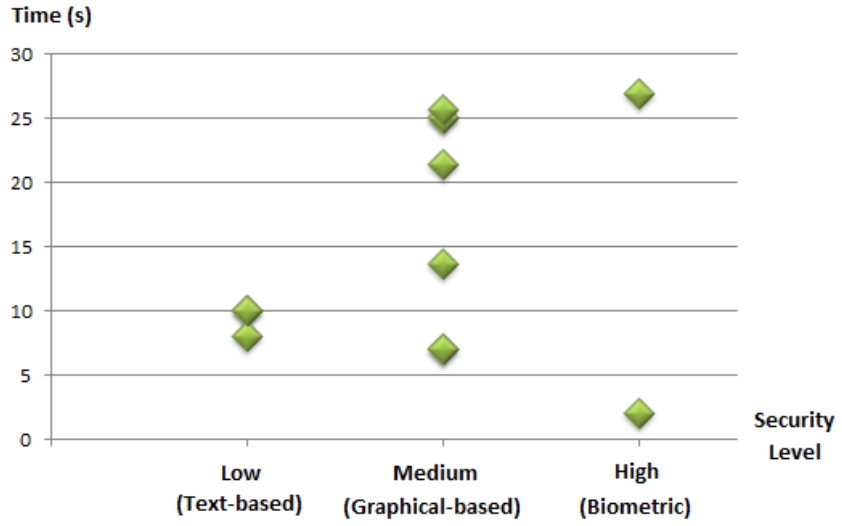

Figure 5.3 Login time by security level

To try to explain this behaviour, we focused on the authentication method. The common factor affecting these methods is the memory needed to recall and recognize the password. To do this, we set three levels:

“Low” for the SSSL and eye tracker methods, because the user does not have to recall the password.

“Medium” for the Color Login method, because this authentication method provides help to the user to recall and recognize the password (through the use of colours).

“High” for the methods where the user needs to recall the password.

Using this categorization, we plotted the chart shown in Figure 5.4.

Table 5.4 User memory load by scheme

Scheme	Shoulder Surfing Safe Login (SSSL)	Text	Mnemonic	Graphical (Recognition - Based)	Graphical (Click-based & Choice-based)	Color Login Graphical password	Graphical based Hybrid Solution (recogniton & recall)	Click & Draw - based Graphical Password (CD-GPS)	Draw a Secret (DAS) Graphical	Click Buttons according to Figures in Grids (CBFG)	Mouse Dynamics Biometrics (Draw Gestures)	Eye Tracker
Login Time	8	10	10	25	25	7	7	13.7	25.7	21.4	26.9	2
User Memory Load	Low	High	High	High	High	Medium	High	High	High	High	High	Low

The chart illustrated in Figure 5.4 shows that efficiency was best (low login time) when the user did not have to rely on memory to recall the password (for example, eye tracker, where there is no password). The methods that help users to recognize the password (for example, shoulder surfing safe login, SSSL) also achieve a good login time.

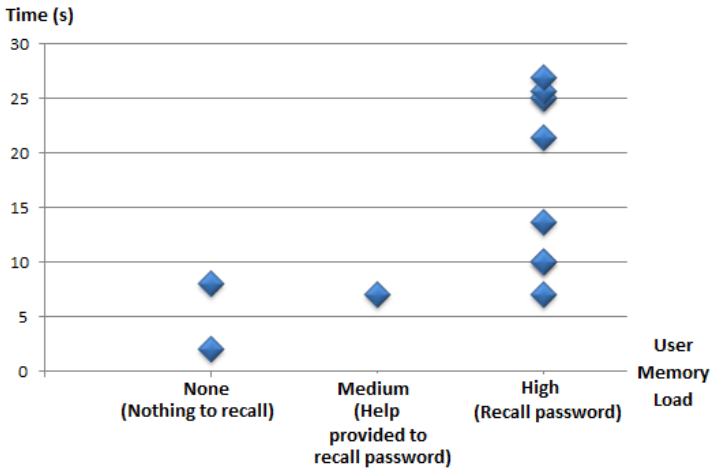

Figure 5.4 Login time by memory load

Efficiency during authentication does not depend on the security level, because the login time is good with high, medium and low security.

"Reduce memorability load" is one of the usability principles stated by Allen et al. (1997). We consider that high login times are caused by violations of this principle by the implementations of these methods.

5.3.4 Threats to Validity

Until we find other systematic reviews focusing on the interaction between efficiency (as part of usability) and authentication (as part of security), we will not be able to validate our study externally.

Regarding internal validity, the author and supervisors were involved in this systematic mapping study. We discussed and agreed on the procedure and considered activities to counteract the effect of researcher bias. On the other hand, we used general terms and placed no constraints on the search strings in order to achieve better coverage as well as high accuracy. We selected three of the most important electronic data sources to which we had access, and added other external sources. The chosen time-frame was intended to include the last decade of research.

During the exclusion process, we were particularly careful not to discard any potentially interesting paper. For this reason, we also included papers whose abstract or title was not completely clear with respect to our research question for further reading.

It was also rather difficult to distinguish the research focus of some papers. So, even though we agree on the result of this process, replicating authors might categorize the studies differently.

The papers do not directly answer the research question, and we had to identify the response by analysing and inferring what authors had found in their research. However, we negotiated this process.

5.4 Conclusions

The combination of traditional authentication schemes (such as user-password, graphical-based, and biometric systems) is generating “new” authentication methods, which try to offset the drawbacks of other methods. This is becoming a patch rather than a thorough-going solution, which makes user authentication harder.

Authentication efficiency does not depend on the security level, because a high, medium and low security level can all have high user efficiency. In other words, efficiency could be good using the most secure authentication method, although login time was low in all cases for text-based authentication.

We found a relationship between efficiency and the memory load, but this requires further analysis.

References

- Allen, B., Brown, A., Eckols, S., Jania-Smith, D., Jhones, S., Levine, S., Sheldon, K. (1997). Handbook of Usability Principles.Center for Learning, Instruction, & Performance Technologies San Diego State University

- Chowdhury, S. Poet, R. (2011). Comparing the Usability of Doodle and Mikon Images to be Used as Authenticators in Graphical Authentication Systems. 2011 International Conference on User Science and Engineering (i-USEr).

- Dailymail (2012).The most common passwords used online in the last year revealed (and 'password' STILL tops the list). [Online] Available from: http://www.dailymail.co.uk/sciencetech/article-2223197/Revealed-The-common-passwords-used-online-year-password-STILL-tops-list.html
[Accessed 15th May 2013]

- Eljetlawi, A.M. (2010). Graphical password: Existing recognition base graphical password usability. Networked Computing (INC), 2010 6th International Conference on.

- Goyal, P.; Bansal, N., Gupta, N. (2013). Averting man in the browser attack using user-specific personal images.Advance Computing Conference (IACC), 2013 IEEE 3rd International.

- Han, W., Cao Y., Lei, C. (2011). Using a Smart Phone to Strengthen Password-Based Authentication. Internet of Things (iThings/CPSCom), 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing.

- Helkala, K. (2012). Disabilities and Authentication Methods: Usability and Security. 2012 Seventh International Conference on Availability, Reliability and Security.

- Hentati, R.; Dorizzi, B.; Aoudni, Y.; Abid, M. (2012). Measuring the Quality of IRIS Segmentation for Improved IRIS Recognition. Signal Image Technology and Internet Based Systems (SITIS), 2012 Eighth International Conference on.

- Jali, M. Z.; Furnell, S.M.; Dowland, P.S. (2011). Multifactor graphical passwords: An assessment of end-user performance. Information Assurance and Security (IAS), 2011 7th International Conference on.

- Kroeze,C.; Olivier, M.S. (2012). Gamifying authentication. Information Security for South Africa (ISSA), 2012.

- Meng, Y. (2012). Designing Click-Draw Based Graphical Password Scheme for Better Authentication. Networking, Architecture and Storage (NAS), 2012 IEEE 7th International Conference on.

- Mock K., Hoanca B., Weaver J., Milton M. (2012). Real-time continuous iris recognition for authentication using an eye tracker. CCS '12 Proceedings of the 2012 ACM conference on Computer and communications security. P. 1007-1009.

- Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M. (2008). Systematic Mapping Studies in Software Engineering.12th International Conference on Evaluation and Assessment in Software Engineering.

- Seffah A., Donyaee M., Kline R. B., Padda H.K., (2006). Usability measurement and metrics: A consolidated model. Software Quality Journal, Volume 14, Issue 2, pp. 159-178.

- Wieringa, R., Maiden, N., Mead, N., Rolland, C. (2006). Requirements Engineering Paper Classification and Evaluation Criteria: a Proposal and a Discussion. Journal of Requirements Eng

- ISO/IEC, 2011, ISO/IEC 25010, Software Product Quality Requirements and Evaluation (SQuaRE) – Quality Models for Software Product Quality and System Quality in use. International Standard. Switzerland.

- ISO 9126-1, 2001, Software engineering – Product quality – Part 1: Quality model. International Standard. Switzerland.

Appendix

A detailed list of the literature reviewed in this article follows:

- Gao, H., Liu, X., Wang, S., Liu, H., Dai, R. (2009). Design and Analysis of a Graphical Password Scheme. Innovative Computing, Information and Control (ICICIC), 2009 Fourth International Conference on.

- Jali, M. Z.; Furnell, S.M.; Dowland, P.S. (2011). Multifactor graphical passwords: An assessment of end-user performance. Information Assurance and Security (IAS), 2011 7th International Conference on.

- Liu X.; Qiu J; Ma L; Gao H; Ren Z. (2011). A Novel Cued-recall Graphical Password Scheme. Image and Graphics (ICIG), 2011 Sixth International Conference on.

- Ma Y; Feng J. (2011). Evaluating Usability of Three Authentication Methods in Web-Based Application. Software Engineering Research, Management and Applications (SERA), 2011 9th International Conference on.

- Meng, Y. (2012). Designing Click-Draw Based Graphical Password Scheme for Better Authentication. Networking, Architecture and Storage (NAS), 2012 IEEE 7th International Conference on.

- Mock K., Hoanca B., Weaver J., Milton M. (2012). Real-time continuous iris recognition for authentication using an eye tracker. CCS '12 Proceedings of the 2012 ACM conference on Computer and communications security, Pages 1007-1009

- Perkovic, T.; Cagalj, M.; Rakic, N. (2009). SSSL: Shoulder Surfing Safe Login. Software, Telecommunications & Computer Networks, 2009.SoftCOM 2009. 17th International Conference on

- Sayed, B.; Traore, I.; Woungang, I.; Obaidat, M.S. (2013) Biometric Authentication Using Mouse Gesture Dynamics. Systems Journal, IEEE

- Zangooei Toomaj; Mansoori Masood; Welch Ian. (2012). A Hybrid Recognition and Recall Based Approach in Graphical Passwords. OzCHI '12 Proceedings of the 24th Australian Computer-Human Interaction Conference. Pages 665-673

6 Conclusions

6.1 Overall Results

A single consolidated model specialized in usability was defined based in the unification of some well-known standards and models.

In order to find the nature of the relationship between security and usability, we conducted a systematic mapping of the literature published over the last decade. The major finding was that there is no unanimous answer regarding the nature of the link. However, authors mentioned the following relationships types: inverse, direct, relative, one-way inverse and no relationship.

The inverse relationship is the most often mentioned connection, followed by the direct, and relative relationships, where the least supported associations are one-way inverse and no relationships.

As we did not find a unanimous answer about the nature of the link between security and usability, we conducted a systematic mapping regarding the nature of the relationship between two subfactors: authentication from a well-known consolidated security model provided by a partnership institution; and user efficiency from our consolidated usability model.

The major finding was that user efficiency during authentication does not depend on the security level. User efficiency can be good with a high security level (biometric authentication). User efficiency can be good with a medium security level (graphical-based authentication). However, user efficiency is always good with text-based authentication.

We also found that user efficiency and memory load are correlated during authentication, but this requires further analysis.

More research into the other subfactors of the consolidated models is required. Therefore, this research represents the first part of a larger project to develop a model for security and usability.

6.2 Future Work

There are many possible topics that should be pursued to continue this research, because security and usability are composed of subfactors, of which we analysed just one tuple (authentication and efficiency) here.

Nevertheless, the number of possible combinations is huge, because, as stated earlier, security attributes include privacy, attack detection, availability, non-repudiation, traceability, commercial damage and so on, whereas usability attributes cover satisfaction, understandability, learnability, ease of use, attractiveness, and more.

Additionally, future work could involve a further study of the impact of memory load (memorability) on user efficiency in authentication methods.

6.3 Personal Reflections

Personally, the most gratifying part of this research was the systematic mapping exploring the relationship between security and usability as global attributes, because our results have been accepted for publication by the 15th International Conference on Enterprise Information Systems (ICEIS) 2013.

The hardest part of the project was to create a consolidated usability model, because we had to analyse a lot of standards, which is not an easy task.

Regarding the results, these can be used to detect low software quality, for example, you can immediately find out whether or not the quality of the conflicting attribute is going to be improved by measuring the related subattribute's quality. On this ground, we could strike a balance between two conflicting subattributes during design time.

Finally, if all subattributes are properly balanced, there will be a perfect trade-off between overall security and usability.